ANALYSIS OF STOCHASTIC NETWORKS

by

William Walton McDonald

An Abstract of
An Engineering Report Presented in
Partial Fulfillment of the Requirements
for the Degree
Master of Science in Engineering

ARIZONA STATE UNIVERSITY
Department of Industrial Engineering

May 1966

ABSTRACT

William Walton McDonald, Master of Science in Engineering, Arizona State University, May, 1966. Analysis of Stochastic Networks. Major Professor: A. Alan B. Pritsker.

The purpose of this research was to: (1) develop a general purpose GERT Simulation Program, (2) investigate resultant distributions of GERT networks containing all AND nodes to determine validity of the PERT normality assumptions, and (3) investigate analysis of AND nodes through analytical and simulation methods for distribution of the equivalent time parameter.

The Simulation Model is a fast, flexible, user-oriented computer program requiring only one input card for each activity in the network. Five probability density functions are available for use in describing the distribution of activity durations. Program output includes a criticality index on each activity and node analysis on specified nodes, including mean, variance, probability of realization and histograms of node realization times. Various GERT networks were simulated to provide examples of the use of the Simulation Model and verify results through statistical tests.

Analysis of resultant distributions of PERT networks verified the PERT normality assumptions utilizing approximation formulas for mean and variance of activity durations.

The merge bias correction procedure utilized to produce better approximations of network realization times was found not applicable to general solution of GERT networks containing AND nodes.

ANALYSIS OF STOCHASTIC NETWORKS

by

William Walton McDonald

An Engineering Report Presented in
Partial Fulfillment of the Requirements
for the Degree
Master of Science in Engineering

ARIZONA STATE UNIVERSITY
Department of Industrial Engineering

May 19[illegible]5

ANALYSIS OF

STOCHASTIC NETWORKS

by

William Walton McDonald

has been approved

May 1966

APPROVED:

_________________________, Chairman

____

____

____

Supervisory Committee

ACCEPTED:

____

Department Chairman

____

Dean, Graduate College

ACKNOWLEDGMENTS

The author is deeply indebted to Dr. A. Alan B. Pritsker of Arizona State University for his assistance in the form of instruction and guidance, to this research endeavor.

The discussion and ideas of Mr. Larry Fendley were gratefully received, particularly in the area of sampling from the Beta probability distribution.

A very special debt of gratitude is owed to Mrs. Lou Ann Elgas, of Arizona State University for her diligent effort and excellent style in typing this report.

TABLE OF CONTENTS

CHAPTER PAGE

I. INTRODUCTION 1

Statement of the Problem and Goal 1

Value of an Answer 2

Background and Literature Review 3

Organization of the Report 5

II. REVIEW OF THEORY 7

Program Evaluation and Review Technique 7

Graphical Evaluation and Review Technique 8

III. THE SIMULATION MODEL 14

Rules for Establishing the Network Model 15

Subprograms 16

Subroutine BETAD 16

Subroutine SET(JS) 20

Function DIST 25

Subroutine HIST 30

Subroutine CHI 33

Method of Simulation 40

Input - Output 56

IV. APPLICATIONS OF THE SIMULATION 64

Probabilistic Networks 64

Deterministic Networks 76

V. ANALYSIS OF AND NODES 86

CHAPTER PAGE

VI. SUMMARY, CONCLUSIONS AND RECOMMENDATIONS 98

Summary 98

Conclusion 99

Recommendations 100

BIBLIOGRAPHY 101

LIST OF TABLES

TABLE		PAGE
I.	Simulated Criticality Index on Activities in the Research and Development Problem	67
II.	Results of Node Analysis on Research and Development Problem	68
III.	Simulated Criticality Index for Activities of the Bomber Interceptor Problem	73
IV.	Results of Node Analysis on Bomber Interceptor Problem for Probability and Expected Time of Realization	74
V.	Analysis of Time of Realization ($t < 3$), Nodes 5, 6, and 7, of the Bomber Interceptor Problem	76
VI.	Results of the Simulation of the PERT network.	82
VII.	Criticality Index for all Simulation runs of the PERT Network Analysis	84

LIST OF FIGURES

FIGURE PAGE

1. Logical Characteristics of Flowgraph Theory 11
2. Network Reduction Employing the Topological Equation 11
3. Sample Network Containing Feedback Loops 12
4. GERT Node Logic 13
5. Flow Diagram of Subroutine BETAD. 18
6. Flow Diagram of Subroutine SET(JS) 21
7. Example of Drawing an Independent Sample from a Cumulative Distribution Function, F(x) 26
8. Pictorial Representation of the Rejection Technique where R1 and R2 are Random Numbers Placing the Point P in the Rectangle M x b 29
9. Flow Diagram of Function DIST 31
10. Flow Diagram of Subroutine HIST 34
11. Flow Diagram of Subroutine CHI 38
12. Histograms of Subroutine CHI Output and Tabulated Normal Probability Function. 41
13. Flow Diagram of Main Program. 42
14. Flow Diagram of Main Program 43
15. Flow Diagram of Main Program 44
16. Feedback Network 54

17. One Loop Network 55

18. Two Loop Network 55

19. GERT Network for Research and Development Program. 66

20. GERT Network for the Thief of Bagdad Problem 69

21. Histogram of Node 5 Realization Times for Thief of Bagdad Problem 70

22. GERT Network for Bomber Interceptor Problem. 72

23. Histograms from Node Analysis of Bomber Interceptor Problem . . 75

24. PERT Diagram, Form 1 and Form 2. 77

25. Network Realization Times, PERT Network, 1000 Simulation Trials 79

26. Network Realization Times, PERT Network, 1000 Simulation Trials 80

27. Histogram of Form 1 Network Realization Times with Beta Distributed Activities for 9999 Simulation Trials 81

28. Sample Network Utilizing AND Logic Nodes 88

29. PERT Network Utilized in AND Node Analysis 92

30. Histogram of PERT Network Node Realization Times, Nodes 4 and 6 96

CHAPTER I

INTRODUCTION

Stochastic processes are defined as time variant and probabilistic processes. Analysis of networks with stochastic properties has been greatly facilitated through the application of GERT (Graphical Evaluation and Review Technique).[1] This procedure has provided an efficient means of analysis and communication of systems problems. A complete discussion of the historical development and theory of networks may be found in the GERT document.

I. STATEMENT OF THE GOAL AND PROBLEM

The goal is to develop a generalized technique for analysis of complex systems portrayed by stochastic networks. The research will concern two related problem areas.

The problem is to develop a general purpose GERT Simulator to allow analysis of GERT networks through the technique of Simulation. Analysis of resultant distribution of network realization will be facilitated by this simulation program to assist analytical solution. Specific objectives will be to:

1. Accept as input the GERT logic node assignments for a network while allowing a choice of probability density functions and parameters for activities comprising the network.

---

[1] A. Alan B. Pritsker, GERT: Graphical Evaluation and Review Technique, The RAND Corporation, RM-4973-NASA (Santa Monica, April, 1966)

2. Provide as output, a criticality statement on activities leading to realization of the network and analysis of several designated nodes through histograms of node realization times and statements on frequency of node realization.

3. Investigate the resultant distribution of GERT networks with reference to theory developed in the literature, specifically, the normality assumptions pertaining to PERT networks.

Another problem is to investigate analysis of AND logic nodes through the use of analytical and simulation techniques for probability of node realization and the distribution of the equivalent time parameter.

II. VALUE OF AN ANSWER

The answer is of value in that, for moderate to large GERT networks, the computational aspects of analytically determining the distribution of network realization are immense. Simulation of GERT networks provide a fast, efficient means for obtaining the parameters and the shape of the resultant distribution. The criticality of an event contributing to network realization provides greater insight to the inner-workings of the network. The complete GERT network is simulated providing probabilistic statements about the terminating node(s).

The basic PERT assumption of normality of project completion for large networks is a subject of controversy, limiting wide acceptance of this technique. Simulation of various networks under fixed and probabilistic conditions should provide an answer substantuating the basic assumption or rejecting it, and pointing out the need for further research.

Analysis of AND nodes is an area requiring research. The value

associated with attainment of this need will be to expand the scope of application of GERT analysis of complex systems and place the research endeavor one step closer to the goal stated above.

III. BACKGROUND AND LITERATURE REVIEW

In the past, network analysis of operational systems has been concerned with project scheduling type networks (PERT, CPM, etc.) and signal flow graphs.

The PERT type analysis treats the time of each activity or event as a constant or random variable, where all activities contribute to project completion or realization. The technique is based on assumptions restricting analysis to approximations of the system (project) parameters and confining construction of networks to a specific form. An attempt is made to deal with random time variations by replacing these variables with their expected times and reverting to a deterministic problem. Based on these expected times, the duration of the longest path through the network is utilized in making estimates of project completion time. The distribution of completion time is assumed normal by invoking the central limit theorem.[2]

Stating that the above estimate of project completion time is always optimistic, Fulkerson[3] proposed an improved method for calculating an approximation to the expected length of a critical path in a PERT net-

---

[2] Special Projects Office, Bureau of Naval Weapons, Department of the Navy, PERT Summary Report, Phase 1, (Washington, D. C., July, 1958)

[3] D. R. Fulkerson, Expected Critical Path Lengths in PERT Networks, The RAND Corporation, RM-3075-PR, (Santa Monica, March, 1962) p. 1.

work by considering the activity duration times as random variables.

A study by MacCrimmon and Ryavec[4] provided an evaluation of the errors inherent on the basic assumptions through analysis of the minimum and maximum duration time of parallel network paths. Their conclusion was that several independently parallel paths of approximately equal duration was found to give the largest error in PERT calculated mean and variance. Also, cross connected paths with shared activities reduced the errors considerably. The presence of one path of length significantly longer than any of the other paths reduced the errors further.

An even more revealing conclusion was drawn by Charnes, Cooper and Thompson[5] in their critical path analysis of PERT networks, utilizing chance constrained programming methods. Whenever there were parallel paths that alternated in criticality and involved sufficiently different times, they found that multimodality of network distribution times often existed.

Simulation of the basic PERT network was first performed by Van Slyke,[6] through application of the Monte Carlo technique to obtain more valid statistics of system behavior. Activity durations were treated as random variables by assigning probability density functions to the individual activities. A criticality index of the relative frequency an activity appeared on the critical path of a network was presented.

---

[4]K. R. Mac Crimmon and C. A. Ryavec, An Analytical Study of the PERT Assumptions, The RAND Corporation (Santa Monica, December, 1962).

[5]A. Charnes, W. W. Cooper, and G. L. Thompson, "Critical Path Analysis Via Chance Constrained and Stochastic Programming," Op. Res., Vol. 12, No. 3, 1964, pp. 460-470.

[6]R. M. Van Slyke, "Monte Carlo Methods and the PERT Problem," Op. Res., Vol. 11, No. 5, 1963, pp. 839-860.

Flowgraph theory has been applied to the analysis of probabilistic systems, utilizing basic properties of flowgraphs. Realization of the flowgraph network results from consideration of all the transmittances or branches of the graph.[7]

The general concepts and fundamentals presented in GERT provide a convenient basis for analysis of complex systems portrayed by stochastic networks. The research endeavor responsible for the conceptualization of GERT is a continuing process, as the need for a generalized technique to analyze stochastic networks has not been fully realized.

Conceptual and computational problems exist in GERT analysis of stochastic networks. Specifically, no general method of analysis exists for the AND logic node. The need for further research is pointed out by Pritsker,[8] while presenting concepts, and approaches and examples. The use of the AND node in GERT is the same as in PERT, where all activities, or branches entering the node must be realized prior to the realization of a branch emanating from that node.

IV. ORGANIZATION OF THE REPORT

The following chapters of this report are organized in the following manner. Chapter II presents a review of the fundamentals of PERT and GERT to establish a base for succeeding chapters. Chapter III describes the GERT Simulation Model with discussion, flowdiagrams and computer listings. Applications of the Simulation Model follow in Chapter IV, covering

---

[7] Pritsker, op. cit., p. 16.

[8] Ibid., Appendix B.

problems presented in the GERT Manual. Also included, are several variations of a PERT network utilized to test theory developed in the literature. Chapter V presents analysis of AND nodes through analytical and simulation treatments. The summary, conclusions and recommendations, are contained in Chapter VI.

CHAPTER II

REVIEW OF THEORY

The purpose of this chapter is to present the general concepts and theory of PERT and GERT necessary to establish a foundation for the material in the following chapters.

I. PROGRAM EVALUATION AND REVIEW TECHNIQUE

The Program Evaluation Review Technique (PERT) is a method of planning, scheduling, and controlling a project by first defining all significant segments and activities of the project and then constructing an interconnecting network of nodes and arrows depicting the various time and precedence relationships necessary for completion of the project.

The time duration of each activity is estimated by three parameters a, m, and b, where a is the earliest (optimistic), b is the latest (pessimistic), and m is the most likely completion time.

The Beta distribution of the form,

$$f(t) = K\,(t-a)^{\alpha}\,(b-t)^{\gamma} \qquad a < t < b$$
$$= 0 \qquad \text{otherwise}$$

(where K, α, and γ, are functions of a, m, and b) is assumed to represent the distribution of activity duration. A simple linear approximation for activity duration was derived without emperical evidence and is in use today.[1]

[1]Special Projects Office, Bureau of Naval Weapons, Department of the Navy, PERT Summary Report, Phase 1, (Washington, D. C., July, 1958), Appendix B.

$$t_e = \text{expected time of an event}$$
$$= (a+4m+b) / 6$$
$$VAR(t_e) = (b-a)^2 / 36$$

Although these two expressions were obtained from the Beta distribution, the usual procedure is to assume normally distributed activity duration times with the parameters given above.

The time at which a node is realized (achieved) is the maximum of the durations of the inwardly-directed paths to that node, since all of the activities on the paths directed into the node must have been completed Outwardly-directed activities from a node cannot start (be released) until the node is realized. The project duration is then the maximum of the elapsed times along all paths from the origin to the terminal node. The path(s) determining total project duration is designated the critical path.

Solution of PERT networks is accomplished by deriving the critical paths of the network based on the expected times of activity durations. The stochastic element is completely ignored through solution of a deterministic network. The distribution of project or network completion times is assumed normal using the central limit theorem. Probability statements concerning project completion are made from the normal distribution with mean and variance derived from activities on the critical path.

II. GRAPHICAL EVALUATION AND REVIEW TECHNIQUE

The Graphical Evaluation and Review Technique (GERT)[2] is a technique for the analysis of generalized networks embodying probabilistic and

---

[2] A. Alan B. Pritsker, GERT: Graphical Evaluation and Review Technique, the RAND Corporation, RM-4973-NASA, (Santa Monica, April, 1966)

time varying characteristics. The general term associated with an activity in the network is a random variable and a probability of occurrence is associated with the realization (acceptance) of the activity. PERT networks are in a sense GERT networks where the probability of realization of an activity is one. Therefore, all activities in the network must be transversed for network completion to occur.

GERT networks are constructed in the familiar manner as an interconnecting network of nodes and arrows depicting the various time and precedence relationships. However, the form of the nodes indicating logical relationships, and the parameter notation for the activity duration times allow GERT, through use of applicable network algebra, to achieve a significant development in generalizing network analysis.

The dual parameter of time and probability is incorporated into one variable allowing use of applicable network algebra. In describing the time parameter, GERT utilizes the moment generating function of the density function (M.G.F.) of the time to realize an activity. The product of the probability of realizing a given activity and the moment generating function is named the w-function, and is defined as follows:

$w_j(s) \quad = \quad p_j M_{tj}(s)$

$p_j \quad = \quad$ the probability of realizing activity j

$M_{tj}(s) \quad = \quad$ the moment generating function of the time to realize the branch activity j

Flowgraph theory provides a convenient basis for analysis of stochastic networks using the w-function from GERT and the logical characteristics of flowgraphs. The logical characteristics utilized are shown in Figure 1 for the series, parallel, and feedback cases through

application of the law of nodes of flowgraph theory.

Substitution of the w-function for the lower case a, b, and c, in Figure 1 provides GERT a single parameter embodying time and probability characteristics while permitting utilization of the logic of flowgraph theory.

The equivalent w-function for a simple network as discussed provides information such as equivalent probabilities and time, through the following relationships. The equivalent probability is:

$p_j = w_j(0)$, by setting the variable $s = 0$.

The M G F of the equivalent time is given by:

$M_j(s) = w_j(s) \,/\, w_j(0)$.

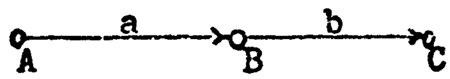
The relationships of network type, equivalent w-function and M G F for the three basic networks is as shown in Figure 2 .[3]

Although most networks can be shown to be combinations of series and parallel equivalent networks, a more orderly approach to network reduction and evaluation is through use of the topology equation of flowgraph theory. The topology equation for closed loops is as follows:

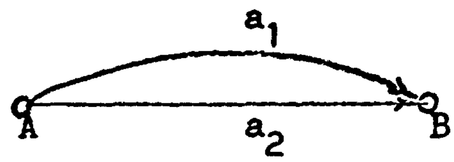
$$H(s) = 1 + \sum_{i=1}^{\infty} (-1)^m L_i(m) = 0 \quad , \qquad m = 1, 2, 3, \ldots\ldots$$

where $L_i(m)$ is the loop product of m non-touching loops. The summation is found of all possible combinations of these loops. Non-touching loops are defined as a series of branches or activities which form loops and the nodes of these branches are non-touching with those of some other loops. Figure 3 illustrates a sample network employing 1st, 2nd, and

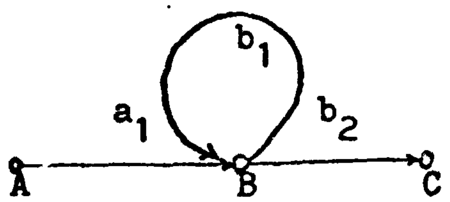
[3]Pritsker, op. cit., p. 28.



a. Series $B = aA \quad C = bB = abA$



b. Parallel $B = a_1A + a_2A$



c. Feedback

$$B = a_1A + b_1B = a_1/(1 - b_1)$$
$$C = b_2B = b_2a_1/(1 - b_1)$$

FIGURE 1

LOGICAL CHARACTERISTICS OF FLOWGRAPH THEORY

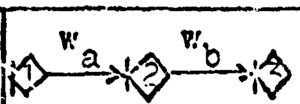
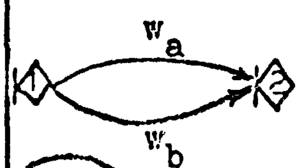
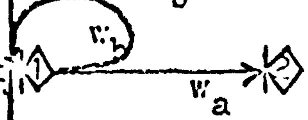
Network Type	Graphical Representation	Paths	Equivalent Function W_e	Equivalent M. G. F. $M_e(s)$
Series		$w_a w_b$	$w_a w_b$	$e^{s(t_a + t_b)}$
Parallel		$w_a ; w_b$	$w_a + w_b$	$1/(p_a+p_b)(p_a e^{st_a}+p_b e^{st_b})$
Feedback		w_a	$w_a/(1-w_b)$	$(1-p_b)e^{st_a}(1-p_b e^{st_b})^{-1}$

FIGURE 2

NETWORK REDUCTION EMPLOYING THE TOPOLOGICAL EQUATION

3rd order loops, where the order of the loop, m, is as above.

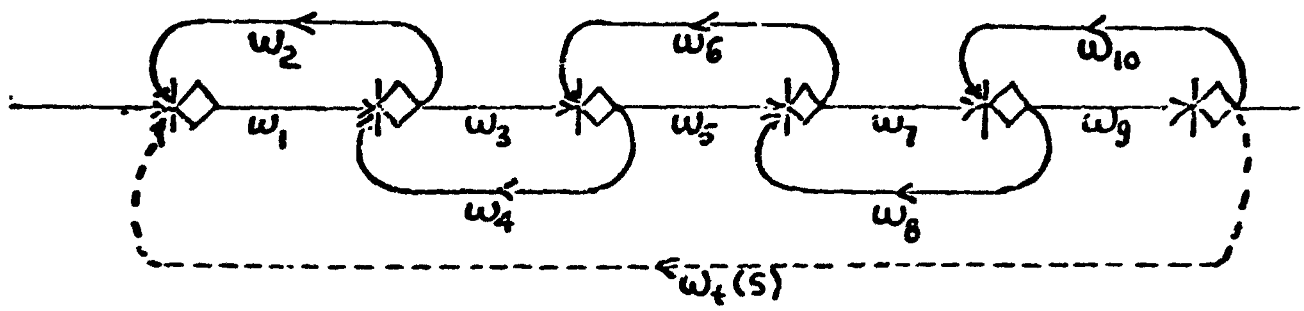


FIGURE 3

SAMPLE NETWORK CONTAINING FEEDBACK LOOPS

There are four loops in Figure 3 of order 1: $L_1(1) = w_1w_2$; $L_2(1) = w_3w_4$; $L_3(1) = w_5w_6$; $L_4(1) = w_7w_8$; and $L_5(1) = w_9w_{10}$. An example of a second order loop is $L_1(2) = L_1(1)\ L_3(1) = w_1w_2w_5w_6$, and a third order loop would consist of $L_1(3) = L_1(1)\ L_3(1)\ L_5(1) = w_1w_2w_5w_6w_9w_{10}$.

The equivalent w-function for a network or section of the network may be obtained by closing the network such as is shown in Figure 3, with the dotted activity from node 1 to node E and assigning the equivalent w-function for the network to this activity, $1 / w_t(s)$. A characteristic of the topological equation is utilized, where for any closed network, $H(s) = 0$, for all s.

Alternatively, $w_t(s)$ could have been obtained directly through application of Mason's rule or the loop rule for open paths.

$$w_t(s) = \frac{\sum(\text{path i})\ (1 + \sum_m (-1)^m\ (\text{loops of order m not touching path i})\ )}{(\ 1 + \sum_m (-1)^m\ (\text{loops of order m})\ )}.$$

GERT utilizes node shapes to indicate logical relationships of activities terminating and emanating at a node. Three types of input node

and two types of output nodes are considered. Figure 4 presents a summary of the GERT node logic for input and output types of nodes. Also included, are the six possible combinations of input-output logic nodes utilized in constructing GERT networks.

The discussion in this chapter was intended to present the basic concepts pertaining to PERT and GERT. GERT is a general network evaluation technique encompassing the principles of PERT and Flowgraphs. While assumptions are inherent in the solution of PERT networks, GERT provides a flexible analysis framework applicable to a wide range of problems. Additional treatment of the theory and applications may be found in the GERT document.

	Name	Symbol	Characteristic
INPUT	EXCLUSIVE-OR		The realization of any branch leading into the node causes the node to be realized; however, one and only one of the branches leading into this node can be realized at a given time.
	INCLUSIVE-OR		The realization of any branch leading into the node causes the node to be realized.
	AND		The node will be realized only if all the branches leading into the node are realized.
OUTPUT	DETERMINISTIC		All branches emanating from the node are taken if the node is realized, i.e., all branches emanating from this node have a p-parameter equal to 1.
	PROBABILISTIC		Exactly one branch emanating from the node is taken if the node is realized.

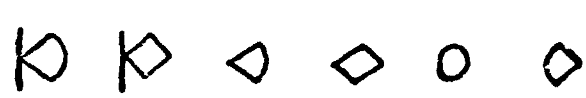
The six possible types of nodes are 

FIGURE 4

GERT NODE LOGIC

CHAPTER III

THE SIMULATION MODEL

The general concepts of GERT provide a convenient basis for analysis of stochastic networks. For modest to large size networks of complex systems, the computational aspects involved in analytically determining the distribution of network realization times are immense. The simulation model, or program, was designed to facilitate analysis of stochastic networks as defined in GERT, and allow as much freedom as possible in depicting networks for analysis.

The purpose of this chapter is to present the construction and operation of the simulation model, deferring applications to a later chapter.

The simulation model can be separated into three distinct parts: (1) Initialization, where the variables are initialized and input data is processed, (2) network simulation, and (3) program output. Five Fortran subprograms, in addition to the main programs are utilized to process input data, assist in the network simulation and output resultant statistics of the simulation trials.

Output from the program includes: (1) A criticality index on each activity as the relative frequency an activity appears on the critical path, (2) the mean and variance of network completion, and (3) analysis of specified nodes, consisting of mean, variance, histogram of node realization times and probability of realizing the node. A Chi Square Goodness of Fit Test may be applied to the distribution of node realization times and printed as output.

This chapter is divided into four sections to facilitate discussion of the model. The sections are: (1) rules for establishing the network model, (2) subprograms, (3) method of simulation, and (4) input-output. Method of presentation is discussion followed by flowdiagram and computer listing where applicable.

I. RULES FOR ESTABLISHING THE NETWORK MODEL

The following rules are necessary to establish restraints on the network model so the computer program will function as intended.

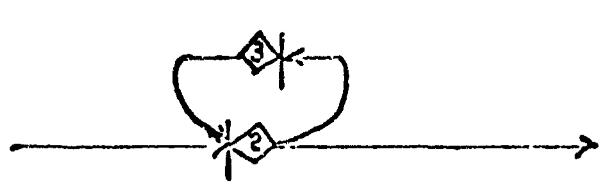
1. There must be one originating node for the network, number 1, although one or more terminating nodes are allowed.

2. Node numbering must be increasing from beginning to end of the network. The I and J node designations, or attributes, of an activity will be numbered such that $J > I$ in the general case.

3. Node numbers may consist of any three digit number, with the exception of 099, which is reserved for input data control.

4. Feedback or looping of an activity to a prior point in the network is indicated whenever the I and J node designations of an activity are such that $J < I$.

5. One exclusive - or node must be placed in the feedback loop when a one-activity feedback, or self-loop, condition exists, to preclude the possibility of $J = I$ as indicated below.



6. The maximum number of activities and nodes is confined to 400 each.

II. SUBPROGRAMS

Five Fortran subprograms account for the greater part of data manipulation and notably simplify the main program logic. They are Subroutine BETAD, Subroutine SET(JS), Function DIST, Subroutine HIST, and Subroutine CHI. This section will present the purpose, theory and discussion of each of these subprograms. Flowdiagrams and computer listing will follow each subprogram.

Subroutine BETAD

The purpose of Subroutine BETAD is to convert the PERT a, m, and b time estimates pertaining to each activity in the network into the parameters α and γ of the Beta probability distribution. The flow diagram is given in Figure 5 and computer listing is found on page 19.

The standardized form of the Beta distribution was previously given as,

$$f(x) = K(x)^{\alpha} \quad (1-x)^{\gamma} \qquad 0 < x < 1$$

$$K, \alpha, \gamma = \text{constants} > 0$$

$$= 0 \qquad \text{elsewhere}$$

Equating expected value and variance of this distribution with the standardized approximations based on the a, m and b time estimates gives the following

$$E(x) = (\alpha + 1) / (\alpha + \gamma + 2)$$

$$= \frac{((a + 4m + b) / 6) - a}{b - a}$$

$$V(x) = (\alpha + 1)(\gamma + 1)/(\alpha + \gamma + 2)^2 (\alpha + \gamma + 3)$$
$$= 1/36.$$

The modal value m, similarly transformed gives

$$r = (m-a) / (b-a)$$
$$= \alpha / (\alpha + \gamma)$$

Elimination of γ from the equation for r and V(x) results in a cubic equation as given below:

$$\alpha^3 + (36r^3 - 36r^2 + 7r)\alpha^2 - 20r^2\alpha - 24r^3 = 0$$

Given $r = (m-a)/(b-a)$, the cubic in α is solved through the Newton Rapheson iterative procedure[1] for - unique value of $\alpha > 1$.

The fact that there is only one positive real root of the cubic equation in α is established through application of Descartes's Rule of Signs.[2]

Successive application of the Newton Rapheson equation,

$$\alpha_{i+1} = \alpha_i - f(\alpha_i)/f'(\alpha_i)$$

results in a very close approximation to the actual value of α sought. The subroutine is designed such that the initial value of α, or α_o, is 100 times the modal value, 100 * r. A maximum of 100 iterations is allowed to achieve an accuracy of at least two decimal places, $f(\alpha_i) \leq 1/100$.

Comparison of calculated values of α and γ with tabulated values

---

[1] Kaiser S. Kunz, Numerical Analysis (New York: McGraw-Hill Book Company, Inc., 1957), p.18.

[2] Ibid., p. 11.

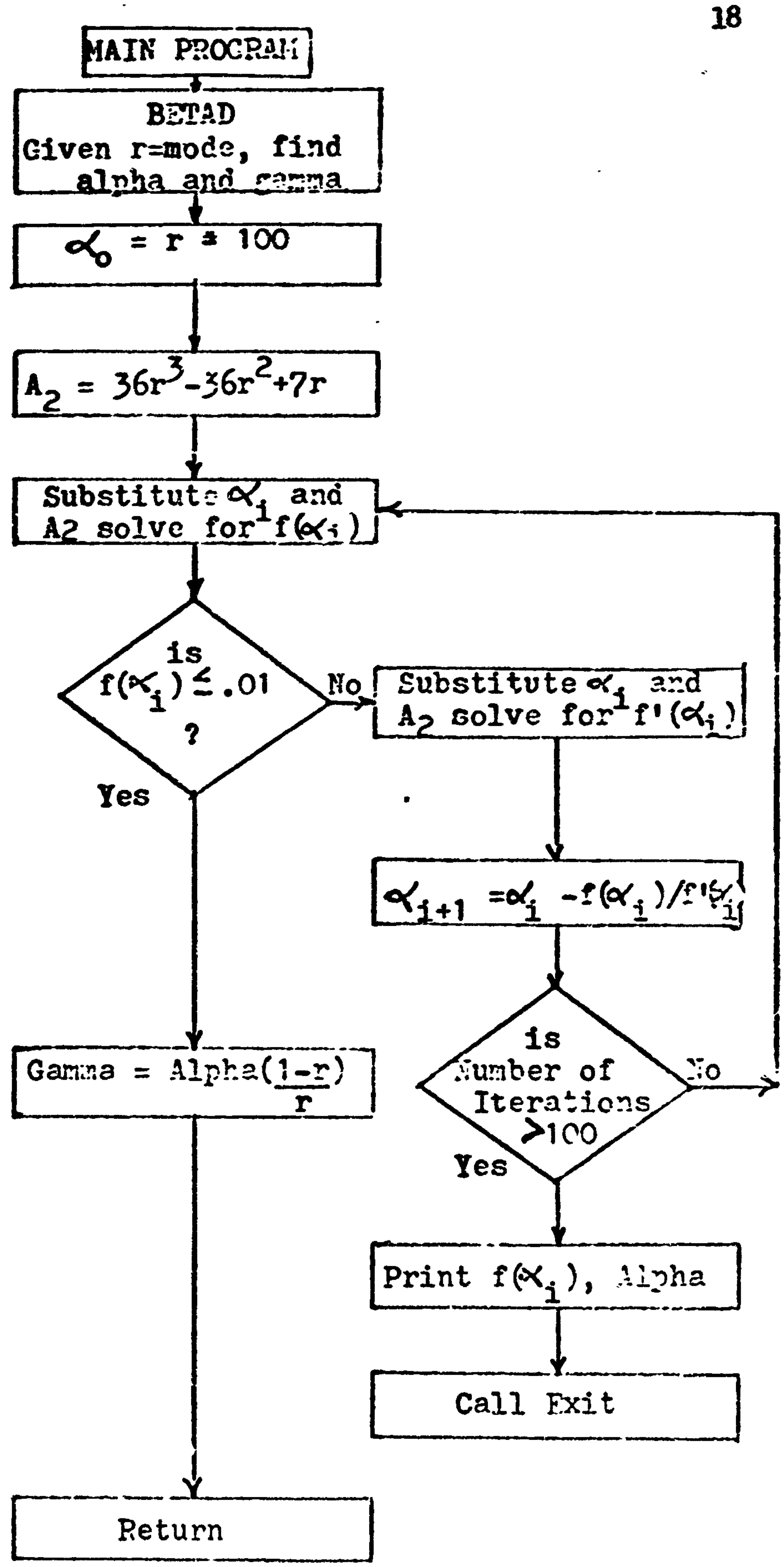


FIG # 5
FLOW DIAGRAM OF SUBROUTINE BETAD

```
      SUBROUTINE BETAD(R,ALPHA,GAMMA)
      DIMENSION A(10)
C
C     START ITERATION WITH SECOND COEFFICIENT
C
      NCOUNT =0
      ALPHA=100.*R
      N=3
      M=1
    5 CONTINUE
      GO TO (6,7,8,9),M
    6 A(1)=36.
      A(2)=-A(1)
      A(3)=7.
      A(4)=0.
      B=R
      X=A(1)
      M=2
      GO TO 10
    7 STORE=X
   12 A(1)=1.
      A(2)=STORE
      A(3)=-20.*R**2
      A(4)=-24.*R**3
      X=A(1)
      B=ALPHA
      N=3
      M=3
      GO TO 10
    8 F=X
      IF(F-.01)11,11,13
   13 A(1)=3.*A(1)
      A(2)=2.*A(2)
      A(3)=-A(3)
      X=A(1)
      N=2
      M=4
   10 DO 20 K=1,N
   20 X=X*R+A(K+1)
      GO TO 5
    9 CONTINUE
      FP=X
      ALPHA=ALPHA-F/FP
      IF(NCOUNT-100)14,14,50
   14 NCOUNT=NCOUNT+1
      GO TO 12
   11 GAMMA=ALPHA*(1.- )/R
      RETURN
   50 PRINT 51,F,ALPHA
   51 FORMAT(12HCURRENT    IS,E12.4,10HWITH ALPHA,F10.4)
      CALL EXIT
      END
```

given by Mac Crimmon and Ryavec,[3] where V(x) = 1/36, indicates very close agreement, as shown below.

		BETAD		REFERENCE	
Mode	Variance	α	γ	α	γ
1/4	1/36	1.22	3.66	1.21	3.63
1/3	1/36	1.82	3.64	1.82	3.64
1/2	1/36	3.00	3.00	3.00	3.00

Subroutine SET(JS)

Subroutine SET(JS)[4] serves the purpose as master time file for ordering activities filed in the array SETS (M, K) as they are released and scheduled for completion in the network simulation cycle. Through the ordering process, Subroutine SET(JS) maintains the system clock pertaining to simulation of the network. The flow diagram is given in Figure 6 and the computer listing on page 22 through 24

As each activity is released and scheduled in the simulation cycle, it is filed in the array SETS (M, K), and Subroutine SET(1) is called. The activities in SETS (M, K) are then ordered by a successor - predecessor arrangement based on column KRANK (1), the column containing the scheduled time of each activity. The three attributes of an acticity stored in SETS (M, K) are the I and J node designations, and t_o, the scheduled time of completion.

This subroutine also provides a marker, MLC (1), signifying the

---

[3]K. R. MacCrimmon and C. A. Ryavec, An Analytical Study of the PERT Assumptions, The RAND Corporation, RM-3408-PR, (Santa Monica, December, 1962), p. 44.

[4]Don Deutsch and Philip Wolfe, "A Revision to GASP, A Simulation Programming Language", (Unpublished term paper, IE 477g, Arizona State University, May, 1965).

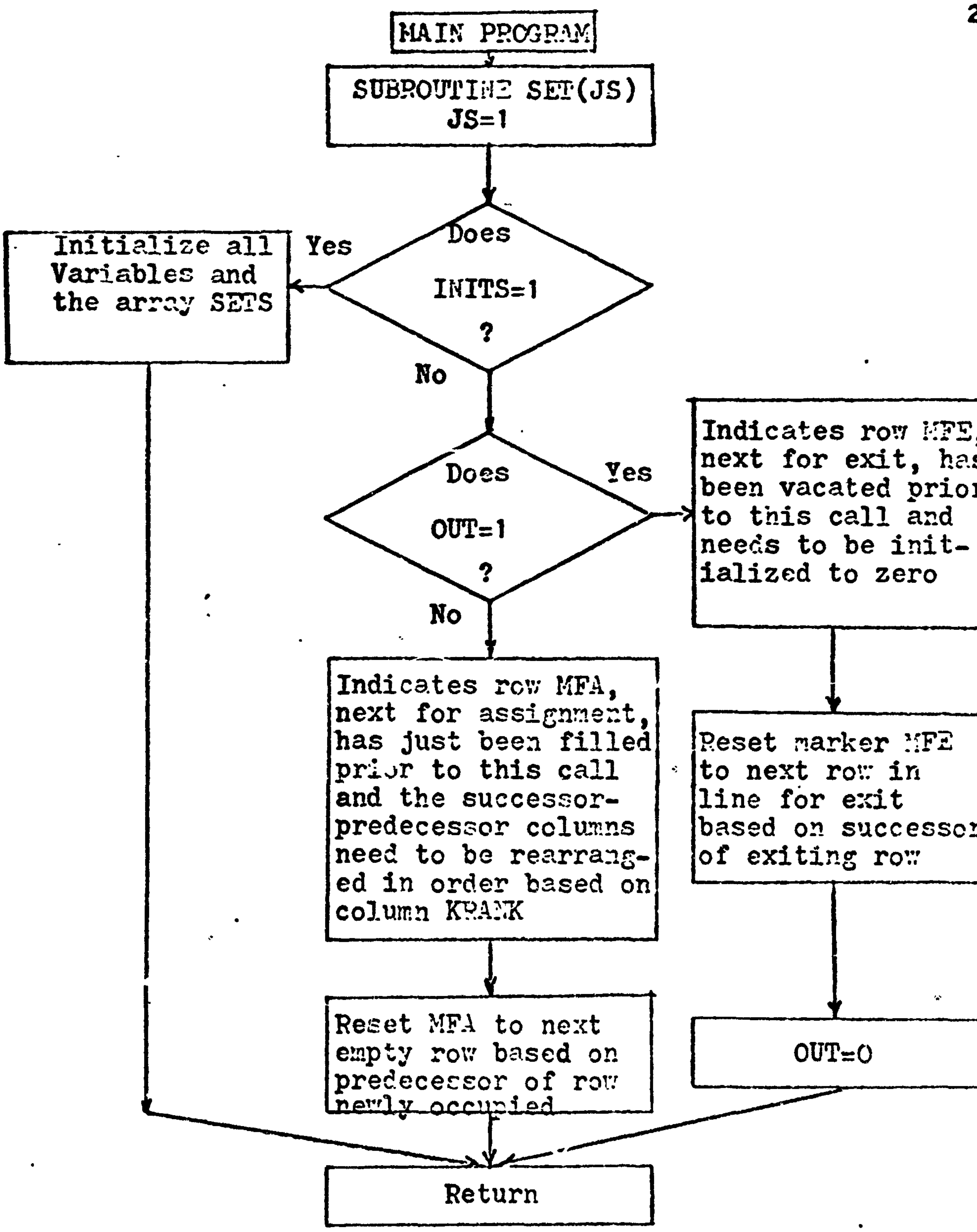


FIGURE 6

FLOW DIAGRAM OF SUBROUTINE SET(JS)

```
      SUBROUTINE SET(JS)
      COMMON RN
      COMMON STATS(1000,4)
      COMMON SETS(40,5),INITS,OUT,KRANK(1),INN(1),MLC(1),MFA
      DIMENSION MFE(1),MLE(1)
C     COMMON SETS,INITS,KRANK,INN,MLC,MFA,MAXNO,QTIME,ENO
C     MUST DIMENSION ALL SUBSCRIPTED VARIABLES IN COMMON BLOCK PLUS
C     THE FOLLOWING VARIABLES,
C     MUST INITIALIZE (JS)****INN(JS)=1 IS FIFO,,INN(JS)=2 IS LIFO
C     MUST INITIALIZE KRANK(JS)
      IF(INITS-1)27,28,27
   28 EOL = 7777.
      ID=20
      NOQ=1
      IM=3
      EOF = 8888.
      WLF = 9999.
      MX = IM+1
      MXX = IM+2
      DO 1 I=1,ID
      DO 2 J = 1,IM
    2 SETS(I,J) = 0.0
      SETS(I,MXX) = I-1
    1 SETS(I,MX) = I+1
      SETS(ID,MX) = EOF
      DO 3 K = 1,NOQ
      MLC(K) = 0
      MFE(K) = 0
    3 MLE(K)=0
      MFA=1
      INITS=0
      OUT = 0.0
      RETURN
   27 MFEK = MFE(JS)
      KNT = 2
      KS = KRANK(JS)
      IF (OUT-1.0) 4,5,100
    4 FA = MFA
      IF(FA-EOF) 8,9,8
    9 PRINT 40
      DO 25 K = 1,ID
   25 PRINT 43, K, (SETS(K,J), J= 1,MXX)
      CALL EXIT
    8 XFA = SETS(MFA,MX)
      IF (INN(JS)-1) 100,7,6
    7 MLEX=MLE(JS)
      IF (MLEX) 100,10,11
   10 SETS(MFA,MXX) = MLE
      MFE(JS) = MFA
   17 SETS(MFA,MX) = EOL
      MLE(JS) = MFA
   14 MFA = XFA
      MLC(JS) = MFE(JS)
      RETURN
   11 IF(SETS(MFA,KS)-SETS(MLEX,KS)) 12,13,13
   13 MSH = SETS(MLEX,MX)
```

A

OCK PLUS

IS LIFO

```
      SETS(MLEX,MX) = MFA
      SETS(MFA,MXX) = MLEX
      GO TO (18,17),KNT
   18 SETS(MFA,MX) = MSU
      SETS(MSU,MXX) = MFA
      GO TO 14
   12 KNT = 1
      MLEX = SETS(MLEX,MXX)
      GN = MLEX
      IF(GN-NLE) 11,16,11
   16 SETS(MFA,MXX) = NLE
      MFE(JS) = MFA
   26 SETS(MFA,MX) = MFEX
      SETS(MFEX,MXX) = MFA
      GO TO 14
    6 IF (MFEX) 100,10,19
   19 IF(SETS(MFA,KS)-SETS(MFEX,KS)) 20,21,21
   21 GO TO (22,16),KNT
   20 KNT = 1
      MPRE = MFEX
      MFEX = SETS(MFEX,MX)
      GN = MFEX
      IF(GN-EOL) 19,24,19
   22 KNT = 2
   24 SETS(MFA,MXX) = MPRE
      SETS(MPRE,MX) = MFA
      GO TO (17,26), KNT
    5 OUT = 0.0
      LC = MLC(JS)
      IF (LC) 100,31,30
   31 PRINT 56,JS
      GO TO 9
   30 DO 32 I = 1,IM
   32 SETS(LC,I) = 0.0
      XT = SETS(LC,MX)
      XS = SETS(LC,MXX)
      JK = XS
      JL = XT
      IF (XT-EOL) 33,34,33
   33 IF (XS-NLE) 35,36,35
   35 SETS(JK,MX) = JL
      SETS(JL,MXX) = JK
   37 SETS(LC,MX) = MFA
      MFA = LC
      MLC(JS) = MFE(JS)
      RETURN
   36 SETS(JL,MXX) = NLE
      MFE(JS) = JL
      GO TO 37
   34 IF (XS-NLE) 38,39,38
   38 SETS(JK,MX) = EOL
      MLE(JS) = JK
      GO TO 37
   39 MFE(JS) = 0
      MLE(JS) = 0
      GO TO 37
```

```
100 PRINT 101,JS
101 FORMAT(1H ,14HEXIT FROM SET ,I2,11H AT ST. 100)
    GO TO 9
 40 FORMAT (//24HOVERLAP  SET GIVEN BELOW/)
 43 FORMAT (I4,10F11.4)
 56 FORMAT (//10HERROR  SET,I4,9H IS EMPTY)
    END
```

row containing the activity bearing the lowest scheduled time and next for exit, based on the scheduling rule FIFO. The marker MFA, or next row for assignment, is also provided. Both marker variables are determined within Subroutines SET(1) through the successor-predecessor arrangement mentioned above.

A more complete description on the operation and use of Subroutine SET(JS) may be found in the referenced paper by Deutsch and Wolfe.

Function DIST

The function subprogram DIST is a prime part of the computer program in that the variable DIST takes on a time value as a random variable of the required probability function of an activity. The parameters of the population distribution are defined with each call of the subprogram. The flow diagram of this subprogram is given as Figure 9 on page 29, and the computer listing is on page 32.

A brief review of sampling theory and use of Monte Carlo methods is offered to facilitate description of the probability distributions available for use in the computer program.

Simulated sampling involves replacing the actual universe of items by an assumed theoretical probability distribution, and then sampling from this theoretical population through use of random numbers.

To draw an item at random from a universe described by the probability density f(x), one derives and plots the cumulative density function F(x),

$$Y = F(x) = \int_{-\infty}^{x} f(u)\, d\, u$$

as described in Figure 7.

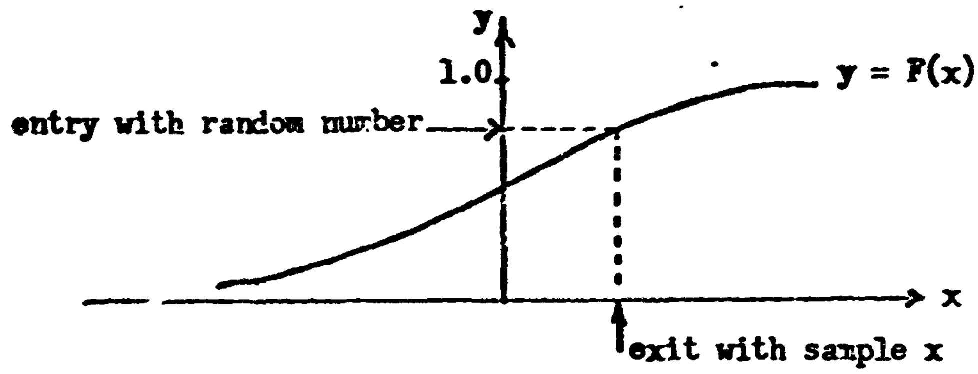


FIGURE 7

EXAMPLE OF DRAWING AN INDEPENDENT SAMPLE FROM A CUMULATE DISTRIBUTION FUNCTION, F(x)

A random number, having equal likelihood of lying anywhere in the interval (0, 1) is selected as the entry on the y axis and projected to an intersection with the curve $y = F(x)$. The value x corresponding to the point of intersection is determined to be the independent sample value of x from the distribution.

With this background in mind, a brief description of the probability distributions available follows.

1. Negative Exponential Distribution

$$y = f(x) = (1/u)e^{-x/u} \qquad x, u > 0$$
$$= 0 \qquad \text{elsewhere}$$
$$F(x) = e^{-x/u} \qquad x, u > 0$$
$$= 0 \qquad \text{elsewhere}$$
$$x = \text{DIST} = -u * \ln(\text{RAND(RN)}),$$

where RAND(RN) is a uniformly distributed random variable and satisfies the relationships: $0 < \text{RAND(RN)} < 1$, and u is the mean of the exponential distribution.

2. Normal Distribution

$$f(z) = (1/\sqrt{2\pi}) \exp(-z^2/2),$$

where $z = (x - u) / \sigma$. Using an approximation to the normal[5], random normal deviates can be obtained from,

$z = (-2 \ln(\mathrm{RAND(RN)}))^{\frac{1}{2}} \cos (2\, \mathrm{RAND(RN)})$.

Thus z is a normally distributed random variable with a mean of 0 and standard deviation equal to 1. Using the transformation,

DIST $= x = \sigma z - u$, a normally distributed variable with a mean of u and standard deviation of σ is obtained. The ratio of σ / u should be less than 1/3 to make the possibility of negative numbers small.

3. Discrete Rectangular Distribution

The discrete rectangular distribution takes on integer values between A and B with equal probability, thus;

DIST = INTF (A +(B - A)*RAND(RN)), where A and B are derived from the relationships $u = (A + B) / 2$ and $\sigma^2 = (B - A)^2/12$.

4. Constant

The time duration of the activity is defined as constant and equal to the value x.

5. Beta Distribution

As described previously, the Beta distribution of the form,

$$y = f(x) = K x^{\alpha} (1-x)^{\gamma} \qquad a < x < b$$

$$= 0 \qquad \text{elsewhere}$$

[5]Claude McMillan, and Richard F. Gonzalez, Systems Analysis, (Homewood, Illinois: Richard D. Irwin, Inc.) p. 157.

has, as its cumulative distribution, an indefinite integral. Tabulated values of the Incomplete Beta Function for limited values of α and γ may be found in Harter.[6]

The method utilized to draw independent samples from the Beta probability distribution is the rejection technique described by Kahn[7], which differs from the sampling procedures utilized for the other probability distributions.

The basic procedure is as follows with reference to Figure 8 . A point is chosen uniformly at random from the rectangle with base of length b and height M. If the point falls below the graph of f(x), accept the abscissa as in independent sample value of the distribution. If not, reject it and try again.

Restating the procedure in step form, and again referring to the graph in Figure 8.

1. Obtain two random numbers, R_1 and R_2.
2. If $R_1 \leq f(a + b\,R_2)\ /\ M$, let $x = a + b\,R_2$.
3. If $R_1 > f(a + b\,R_2)\ /\ M$, pick two new random numbers, R_1 and R_2, and try again.

A programming aspect of this sampling procedure is noted in the seemingly low efficiency of selecting samples from the distribution. Should the range b be very wide, and the value of M very small, the

---

[6] H. Leon Harter, New Tables of the Incomplete Gamma-Function Ratio and of Percentage Points of the Chi-Square and Beta Distribution, Aerospace Research Laboratories, Office of Aerospace Research, United States Air Force (Washington: Government Printing Office, 1964), table 3.

[7] Herman Kahn, Applications of Monte Carlo, The RAND Corporation, AECU-3259, (Santa Monica, April, 1954), p. 10.

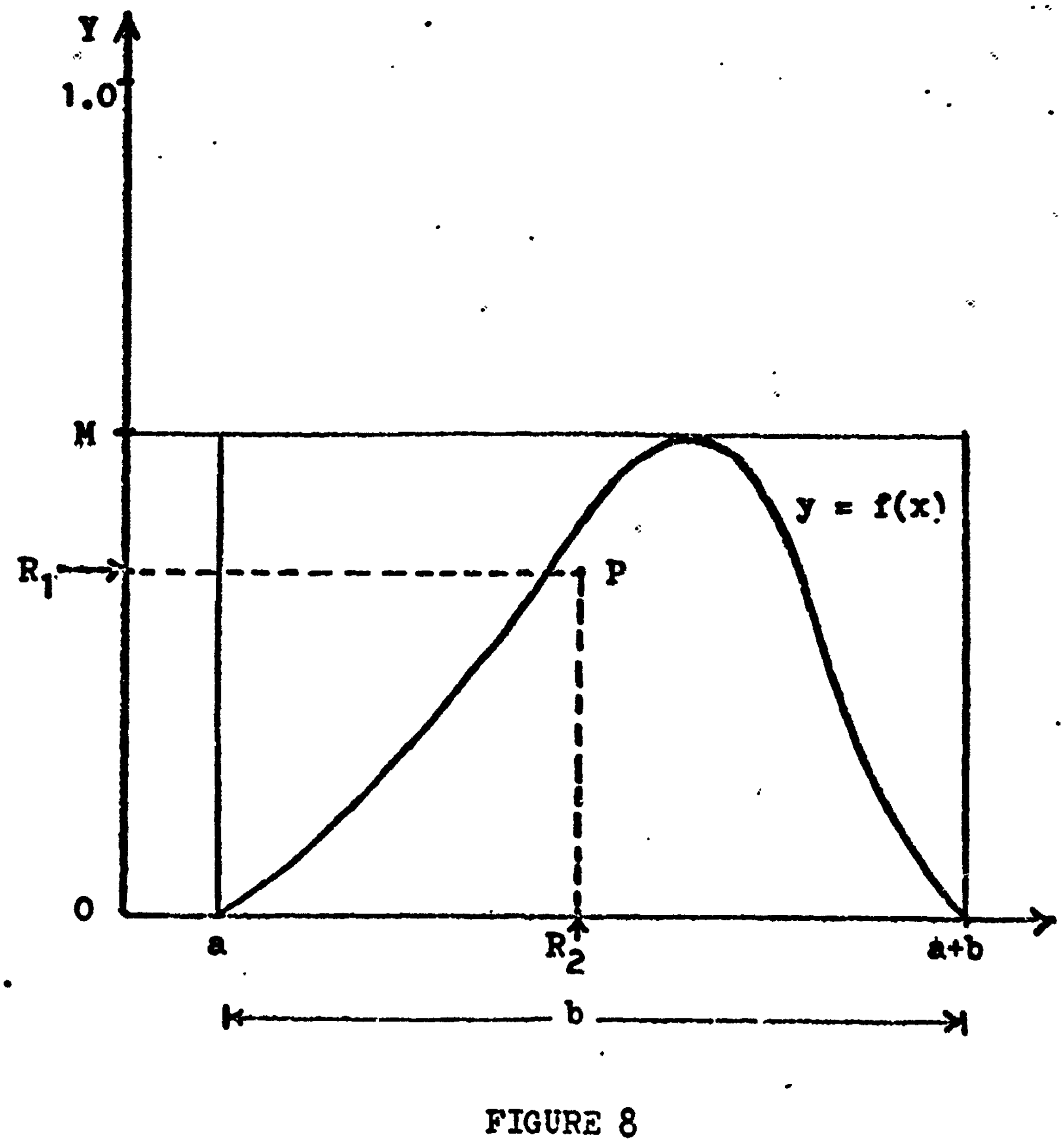


FIGURE 8

PICTORIAL REPRESENTATION OF THE REJECTION TECHNIQUE WHERE R1 AND R2 ARE RANDOM NUMBERS PLACING THE POINT P IN THE RECTANGLE M x b

probability of acceptance of the variable R_2 as an independent sample from the probability distribution would indeed be small, since R_1 is a random number, $0 < R_1 < 1$. Many repetitions of this rejection procedure for each acceptable sample could occur.

A significant improvement in the efficiency of sampling through use of the rejection technique is realized by restricting the random variable R_1 to the range $0 < R_1 < f(M)$, where M is the modal value of the distribution. The rectangle Mxb shown in Figure 8 is then the actual space we utilized.

Although the rejection technique could be utilized in sampling from the other distributions described in this section, the more direct approximations utilized require much less time in computation and are therefore utilized.

A sample network consisting of one activity defined by the PERT time estimates, a, m, and b, of the Beta probability distribution was simulated to test adequacy of the sampling procedure outlined above. The GERT Simulation Program of this report was utilized in simulating the network.

The test was to determine if symmetrical time estimates would produce a reasonable approximation to the normal distribution as determined by the Chi-Square Goodness of Fit Test.

The hypothesis that the sample distribution is normally distributed about its mean was not rejected at the 5% confidence level since the calculated $X^2 = 19.34$ is much less than the critical level of $X^2\ (.95,17) = 27.6$.

Subroutine HIST

Subroutine HIST accomplishes the task of converting the output

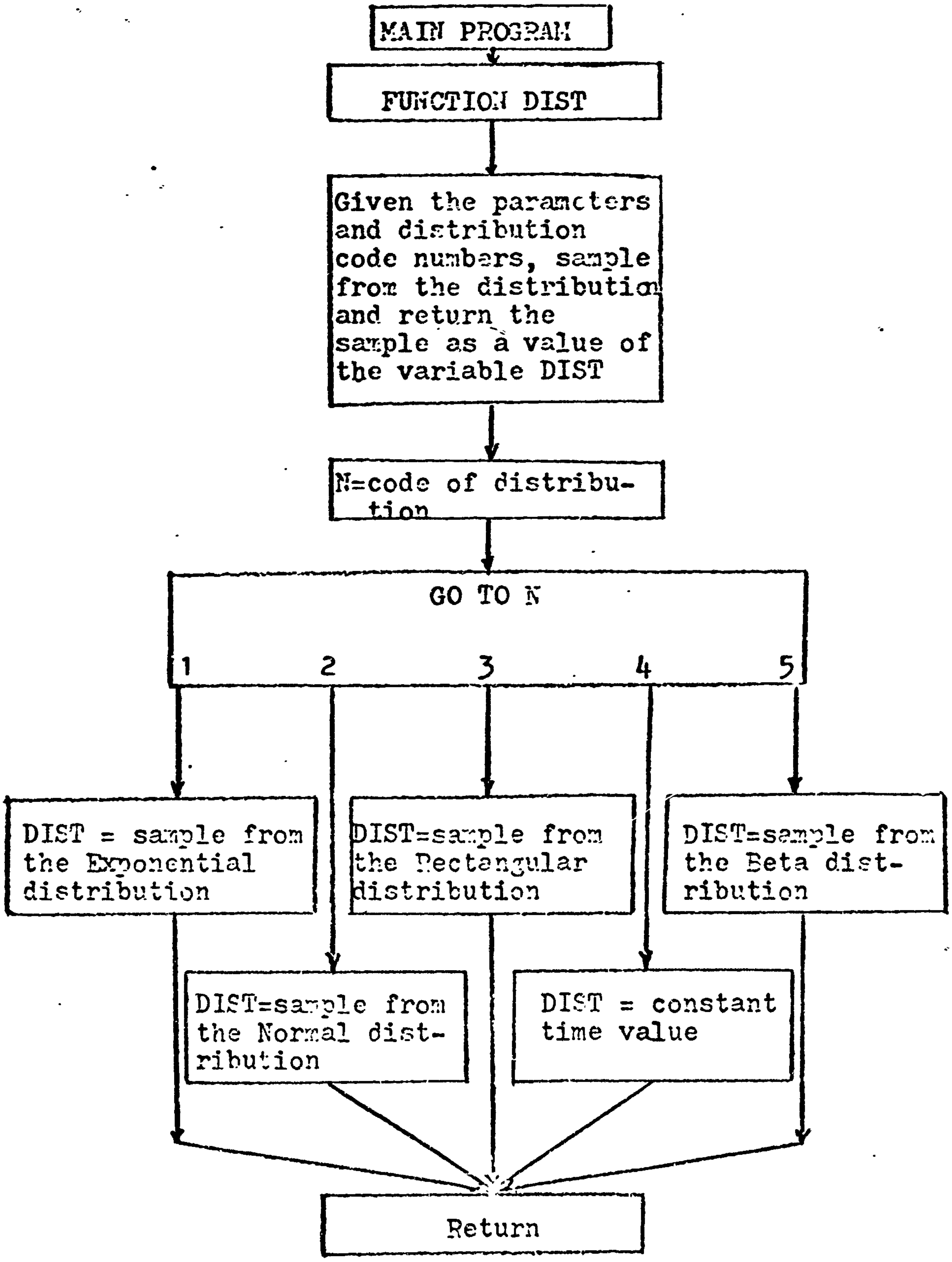


FIGURE 9

FLOW DIAGRAM OF FUNCTION DIST

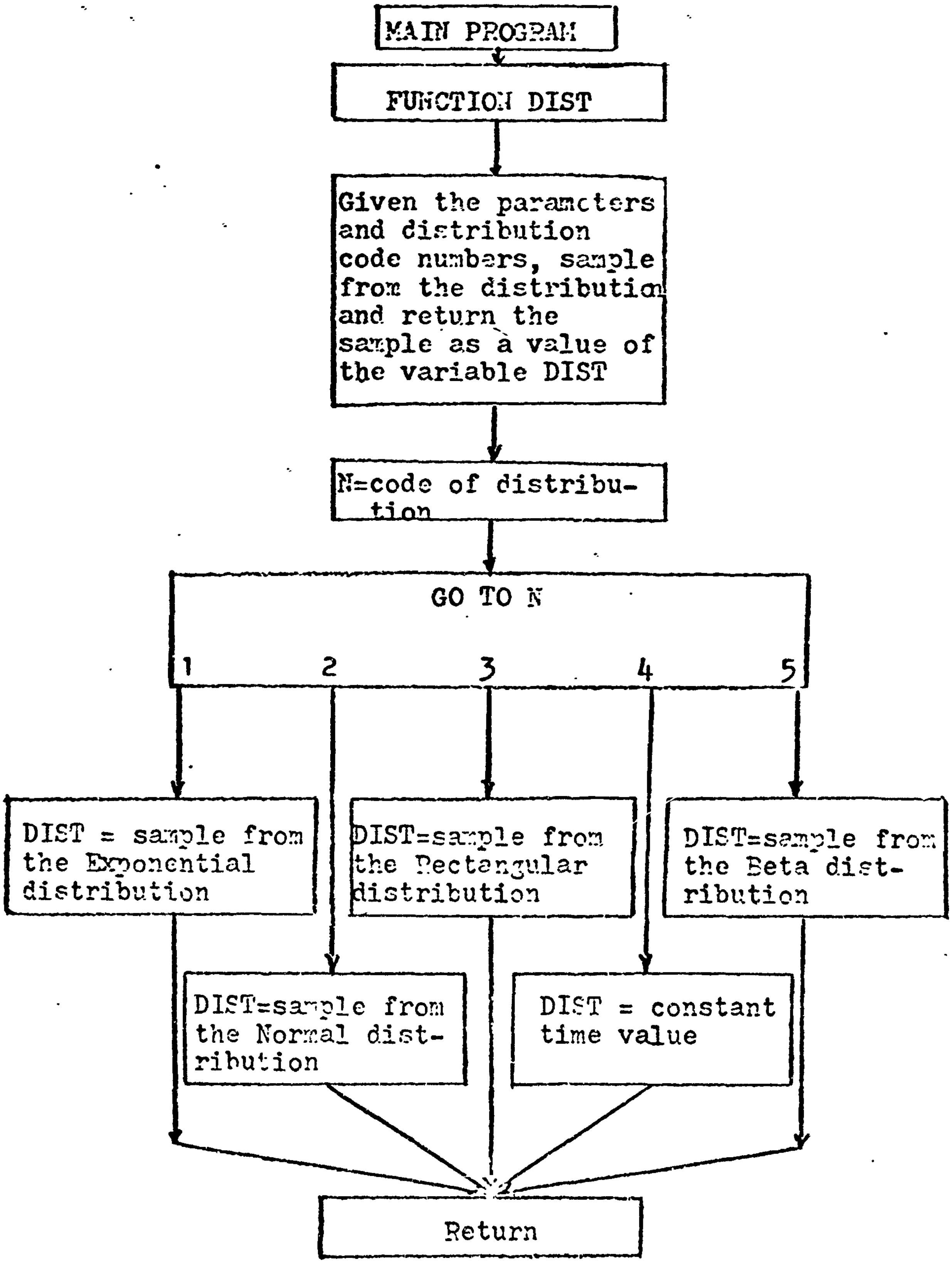


FIGURE 9

FLOW DIAGRAM OF FUNCTION DIST

```
      FUNCTION DIST(X,X1,X2,X3,X4)
      COMMON RN
      COMMON STATS(1000,4)
      COMMON SETS(40,5),UNITS,OUT,KRANK(1),INN(1),MLC(1),MFA
      N=X2
      IF(N)10,10,12
   10 CONTINUE
      DIST=0.0
      RETURN
   12 GO TO (1,2,3,4,5,6,7,8,9,11),N
    1 DIST=-X*LOGF(RAND(RN))
      RETURN
    2 R1=RAND(RN)
      R2=RAND(RN)
      V=(-2.0*LOGF(R1))**0.5*COSF(6.283*R2)
      DIST=V*SQRTF(X1)+X
      RETURN
    3 I=X1+(X-X1)*RAND(RN)
      DIST=I
      RETURN
    5 CONTINUE
       IF(X)17,21,17
   17 IF(X1)18,19,18
   18 CONTINUE
      XMODE=X/(X+X1)
      XMODE=XMODE**X*(1.-XMODE)**X1
   19 T=RAND(RN)
      Y=RAND(RN)
      Y=Y*XMODE
      BETA=T**X*(1.-T)**X1
      IF(BETA-Y)19,20,20
   20 DIST=X3+T*(X4-X3)
      RETURN
   21 DIST=X4
      RETURN
    6 GO TO 4
    7 GO TO 4
    8 GO TO 4
    9 GO TO 4
   11 GO TO 4
    4 DIST=X
      RETURN
   14 FORMAT(6HZERO N//)
      END
```

statistics into a standard histogram configuration of twenty cells. This subroutine requires only identification of the storage array and number of records contained in the array as input. This is a general purpose subroutine with internal decision rules for setting up cell size and location for the most convenient grouping of data and for output. Figure 10 contains the flow diagram of Subroutines HIST and the computer listing is located on page 35.

The data array is reviewed to obtain expected time, variance, high time, low time, and range. Based on these values, the histogram is set up such that, if the range is less than 20, the first cell will be the lowest data value truncated to an integer and cell increments will be of size one. Should the range be greater than 20, the midpoint of the histogram is the expected value, and cell increments will be in integer steps of RANGE/20 + $\frac{1}{2}$.

Output statistics are expected time, variance, high time, low time, range and histogram.

<u>Subroutine CHI</u>

Subroutine CHI is an auxiliary routine used to measure the discrepancies or fit between the simulation output distributions and a theoretical probability distribution. Provision is made only for comparison of symmetrical distributions. Input data is required as described in a later section of this chapter.

This subroutine is utilized only at the analysts discretion by placing the subroutine call card as shown, in the main program listing on page 45, in the histogram output section following the CALL HIST statement. The subroutine flow diagram is given in Figure 11 and the computer listing is on page 39.

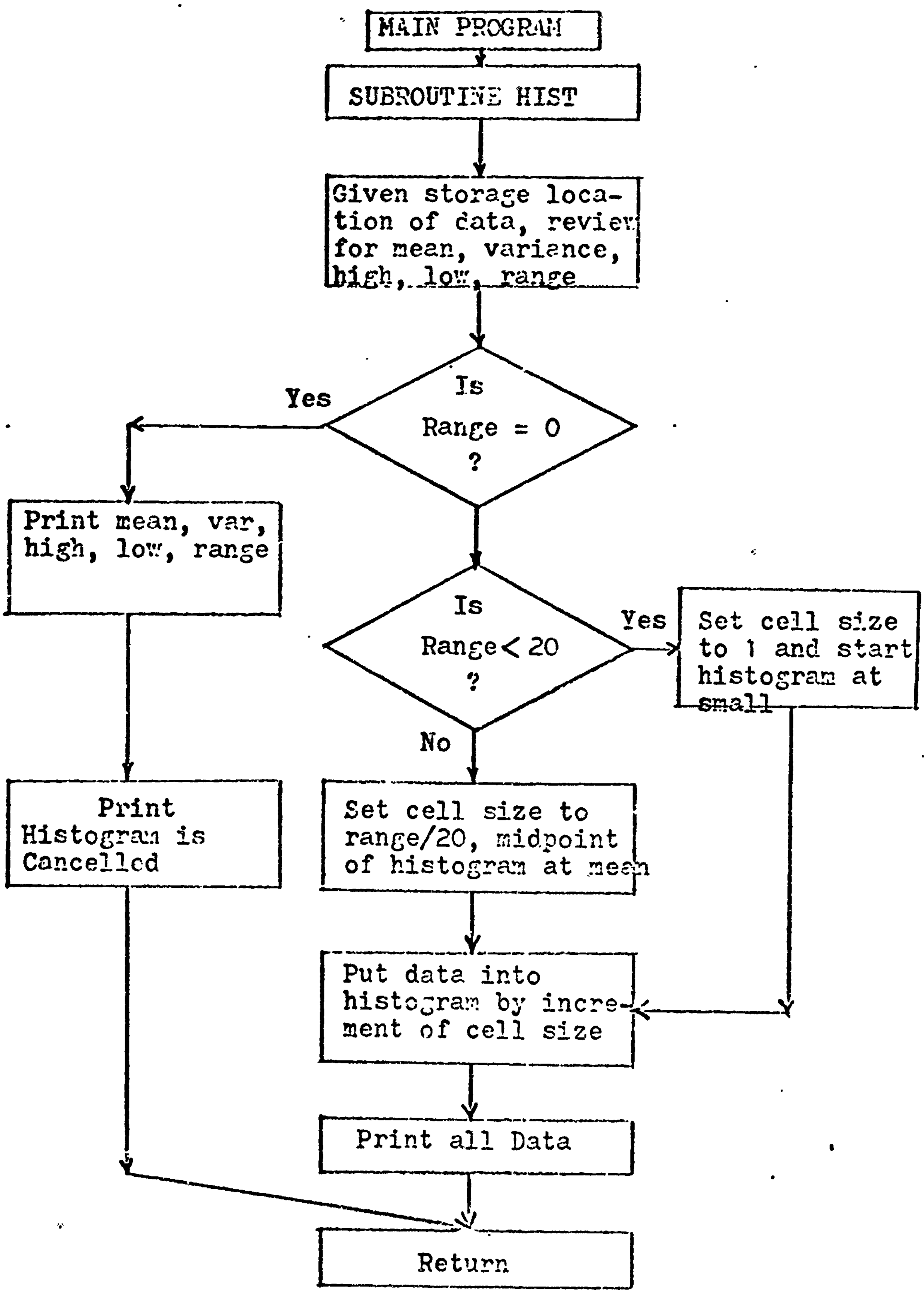


FIGURE 10

FLOW DIAGRAM OF SUBROUTINE HIST

```
      SUBROUTINE HIST(NT,NSIM)
      COMMON RN
      COMMON STATS(1000,4)
      COMMON SETS(40,5),INITS,OUT,KRANK(1),INN(1),MLC(1),MFA
      DIMENSION HISTG(20)
      DO 14 I=1,20
   14 HISTG(I)=0.0
      BIG=-9.0E50
      SMALL=9.0E50
      EXPT=0.0
      VXPT=0.0
      XNSIM=NSIM
      DO 20 I=1,NSIM
      X=STATS(I,NT)
      EXPT=EXPT+X
      VXPT=VXPT+X*X
      BIG=MAX1F(BIG,X)
   20 SMALL=MIN1F(SMALL,X)
      EXPT=EXPT/XNSIM
      VXPT=VXPT/XNSIM-EXPT*EXPT
      PRINT 117,EXPT
      PRINT 118,VXPT
C
      RANGE=BIG-SMALL
      IF(RANGE)80,80,81
   80 PRINT 113
      RETURN
   81 IF(RANGE-20)86,86,87
   86 NSIZE=1
      SIZE=1.0
      NSM=SMALL
      NBG=NSM+19
      ASMALL=NSM
      GO TO 83
   87 SIZE=RANGE/20.
      NSIZE=SIZE
      MSIZE=SIZE+.5
      IF(MSIZE-NSIZE)89,89,88
   88 NSIZE=MSIZE
   89 MIDP=EXPT
      NSM=MIDP-9*NSIZE
      IF(NSM)900,901,901
  900 NSM=0
  901 ASMALL=NSM
      NBG=NSM+19*NSIZE
      SIZE=NSIZE
   83 DO 82 N=1,NSIM
      X=STATS(N,NT)
      IF(X-ASMALL)85,84,84
   84 L=(X-ASMALL)/SIZE+1.
      IF(L-20)82,82,882
  882 L=20
      GO TO 82
   85 L=1
   82 HISTG(L)=HISTG(L)+1.0
      PRINT 115,(I,I=NSM,NBG,NSIZE)
```

```
      PRINT 119,(HISTG(L),L=1,20)
      PRINT 116,BIG,SMALL,RANGE
      RETURN
  113 FORMAT(//29X,22HHISTOGRAM IS CANCELLED)
  114 FORMAT(1H1)
  115 FORMAT(/20I4)
  116 FORMAT(5X,3HBIG,7X,5HSMALL,5X,5HRANGE/3F10.3)
  117 FORMAT(//20X,40HEXPECTED TIME OF COMPLETION THIS NODE IS,F10.
  118 FORMAT(/,20X,40HVARIANCE TIME OF COMPLETION THIS NODE IS,F10.
  119 FORMAT(/20F4.0)
      END
```

A measure of the discrepancies existing between observed and expected frequencies is given by the X^2 statistic where the data is grouped into cells or events of the total sample population. For Large N, or sample size, the sampling distribution X^2 closely approximates the Chi Square probability distribution. X^2 is given as,

,F10.
,F10.

$$X^2 = \sum_{j=1}^{k} (0_j - e_j)^2 / e_j, \text{ with } k\text{-}3 \text{ degrees of freedom.}$$

Each summation represents the observed, 0_j, and expected, e_j, frequencies of occurrence within a cell, where there are k cells.

Expanding and simplifying the above expression where $N = \sum_{j=1}^{k} 0_j = \sum_{j=1}^{k} e_j$, gives the following expression:

$$X^2 = \left(\sum_{j=1}^{k} 0_j^2/e_j \right) - N.$$

This subroutine requires the location of the data storage array, the number of records contained in the array, and whether or not to read input-data describing the theoretical probability density function.

The data in the storage array is reviewed for calculation of the expected time and variance and then transformed through the normalizing equation $z = (x - E(x)) / \sigma_x$. As the data is normalized, it is placed in a 20 cell histogram with center, $z = 0$; variance, $V(z) = 1$; and increments of 3/10. Therefore, the end cells of the histogram represent $\pm 3\sigma_x$. The grouped histogram cells are utilized in calculating the value of X^2.

Output of the subroutine includes the value of X^2 calculated, expected time, variance, and the normalized histogram of data.

Tests of Hypothesis may be made at the significance level desired, with 17 degrees of freedom, based on the tabulated values of the Chi Square probability density function.

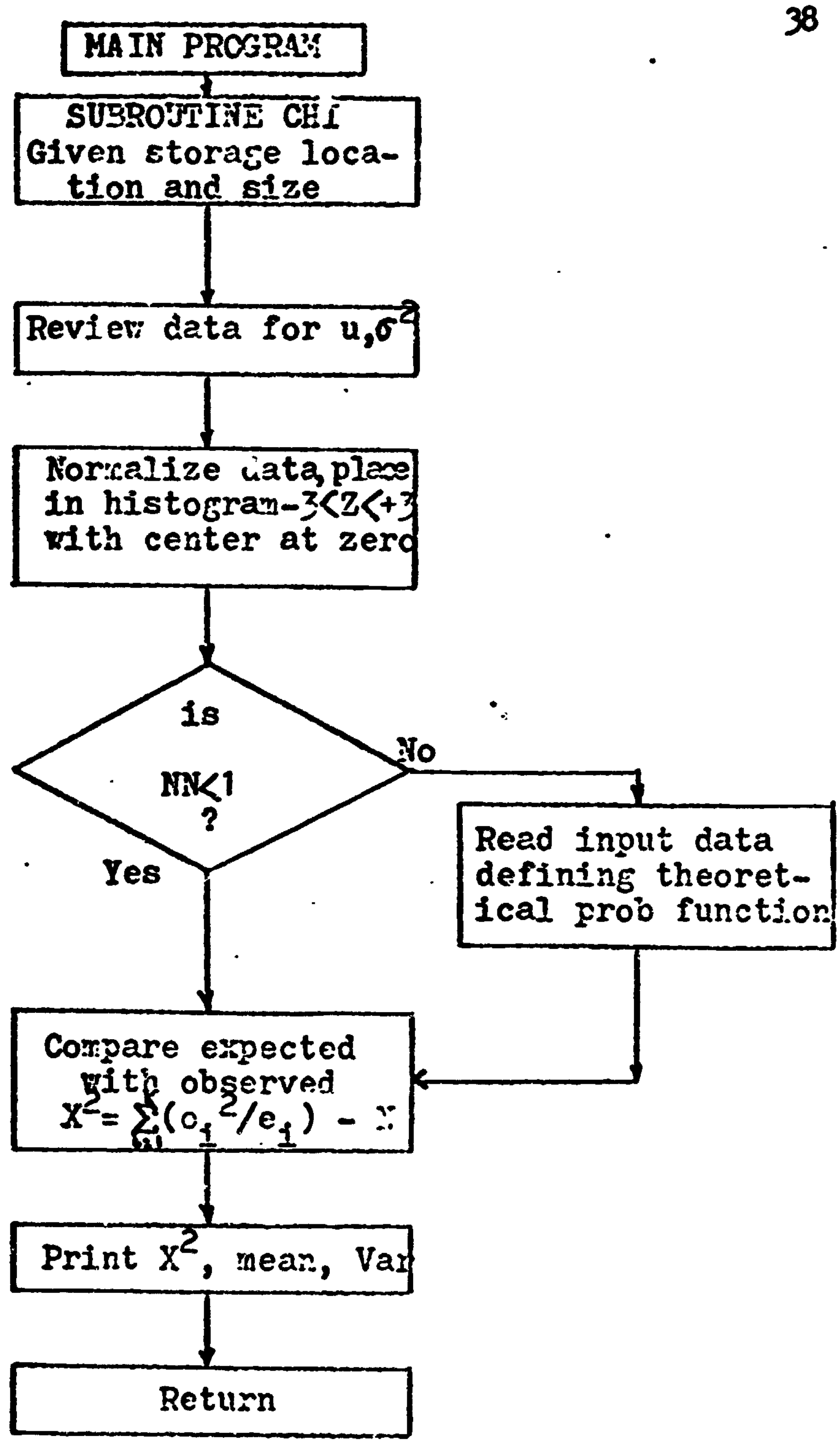


FIGURE 11

FLOW DIAGRAM OF SUBROUTINE CHI

```
      SUBROUTINE CHI(NT,NSM,NN)
      COMMON RN
      COMMON STATS(10000,1)
      DIMENSION HIST(20),DATA(20)
      DO 2 I=1,20
    2 HIST(I)=0.0
      VAR=0.0
      SUM=0.0
      XCHI=0.0
      DO 10 I=1,NSM
      X=STATS(I,NT)
      SUM=SUM+X
   10 VAR=VAR+X*X
      XNSM=NSM
      ETIME=SUM/XNSM
      VAR=VAR/XNSM-ETIME*ETIME
      SIG=SQRTF(VAR)
      DO 20 I=1,NSM
      X=STATS(I,NT)
      ICODE=1.0
      Z=(X-ETIME)/SIG
      IF(Z)5,7,7
    5 ICODE=0.0
    7 N=ABSF(Z)*10./3.+1.
      IF(N-10)12,12,11
   11 N=10
   12 L=N+10*ICODE
      IF(L-10)8,8,9
    8 L=11-L
    9 HIST(L)=HIST(L)+1.
   20 CONTINUE
      IF(NN-1)22,22,41
   22 DO 40 I=1,10
      READ 66,DATA(I)
      DATA(21-I)=DATA(I)
   40 CONTINUE
   41 XCHI=0.0
      M=0
      DO 60 I=1,20
      XCHI=(HIST(I)**2)/(DATA(M+I)*XNSM)+XCHI
   60 CONTINUE
      XCHI=XCHI-XNSM
      PRINT 67,XCHI
      PRINT 70,ETIME
      PRINT 71,VAR
      PRINT 69
      PRINT 68,(L,L=1,20)
      PRINT 77,(HIST(L),L=1,20)
      RETURN
   66 FORMAT(F6.4)
   67 FORMAT(/16X,41HCHI SQUARE VALUE FOR THIS DISTRIBUTION IS,F10.4/)
   68 FORMAT(20I4)
   69 FORMAT(16X,48HHISTOGRAM OF NORMALIZED DATA, Z=0 AT 11 ON SCALE/)
   70 FORMAT(26X,17HEXPECTED VALUE IS,F10.4/)
   71 FORMAT(26X,11HVARIANCE IS,6X,F10.4/)
   77 FORMAT(20F4.0)
  998 CONTINUE
      END
```

Figure 12 contains an example of the subroutine output, histogram of data input representing the normal probability density function, and histogram of the normalized data.

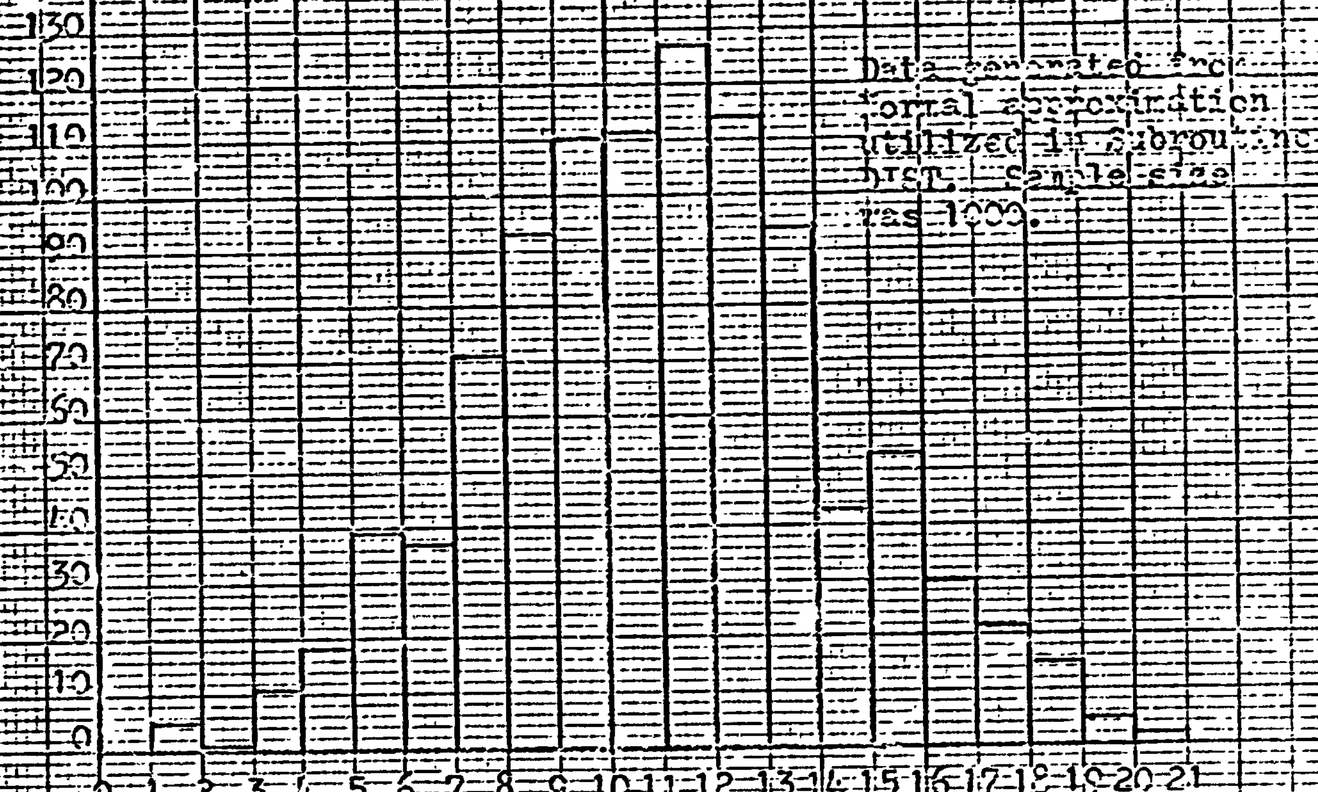
III. METHOD OF SIMULATION

This section will pertain primarily to the logic and procedures utilized in simulating the network model. Input data requirements and description of output is deferred to the following section, Input-Output. The three parts of the simulation model, or program will be discussed with reference to the subprograms previously mentioned. The flow diagrams are given as Figures 13 through 15 and the computer listing of the main program is found on pages 45 through 51.

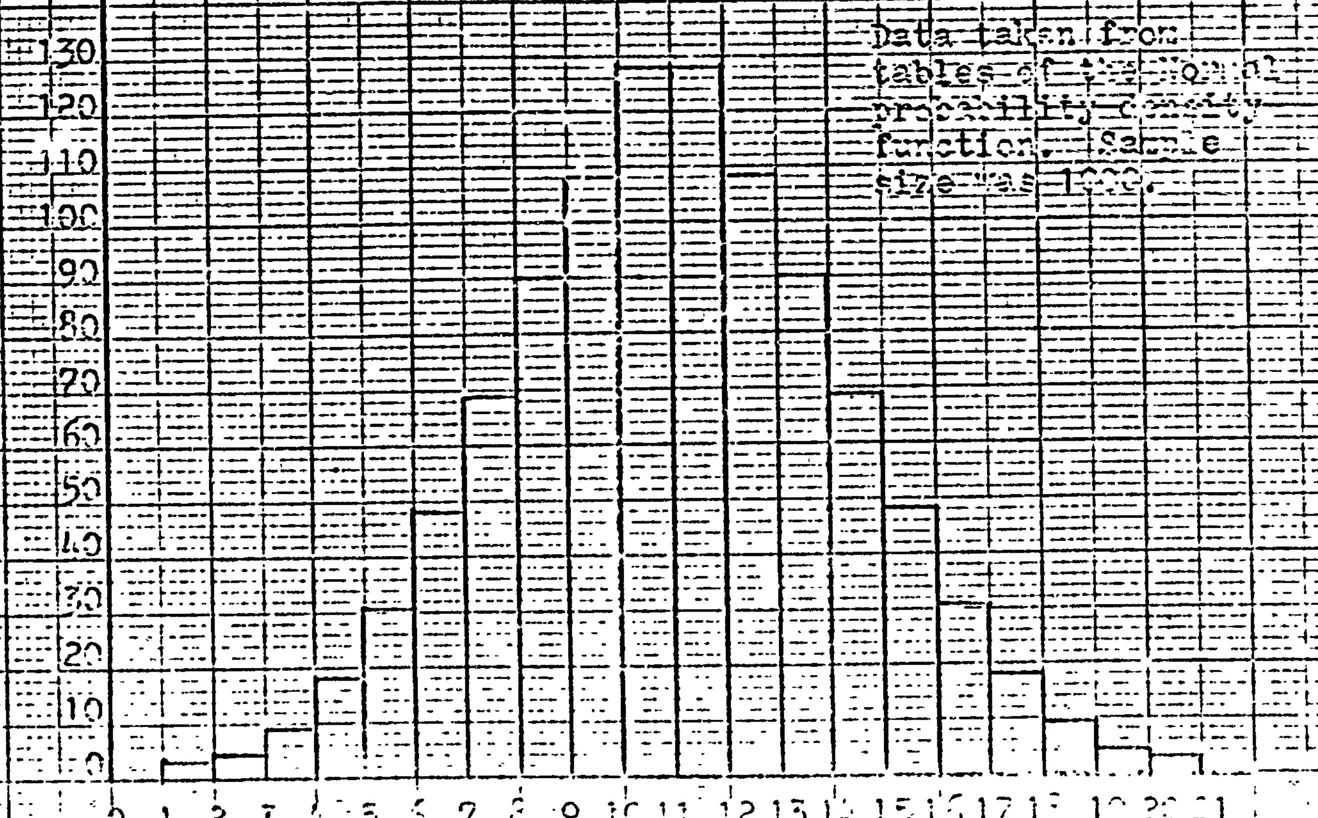
All arrays, variables, and sums are initialized at the start of the program. The random number seed, RN, and number of simulation trials NSIM, are defined by the first network data card read into the program. Should RN be zero the program is terminated.

Activities specifying the network model and their attributes are read into the program and stored in the array STORE(N, I). Attributes of each activity are I-J node designations, codes for the probability density functions, parameters of the density function, and probability of traversal given that the I node has been realized. Subroutine BETAD is utilized to transform the PERT a, m, and b time estimates into parameters α and γ of the Beta probability function when required. As the activities are read into the program the number of activities emanating from a specific node, NQ(I), is calculated.

Nodes and their logic related code are defined on the next input cards and stored as values of the variable NNODE(J). The node information is then printed as output.



HISTOGRAM OF SUBROUTINE CHI OUTPUT



HISTOGRAM OF TABULATED NORMAL PROBABILITY FUNCTION

FIGURE 12

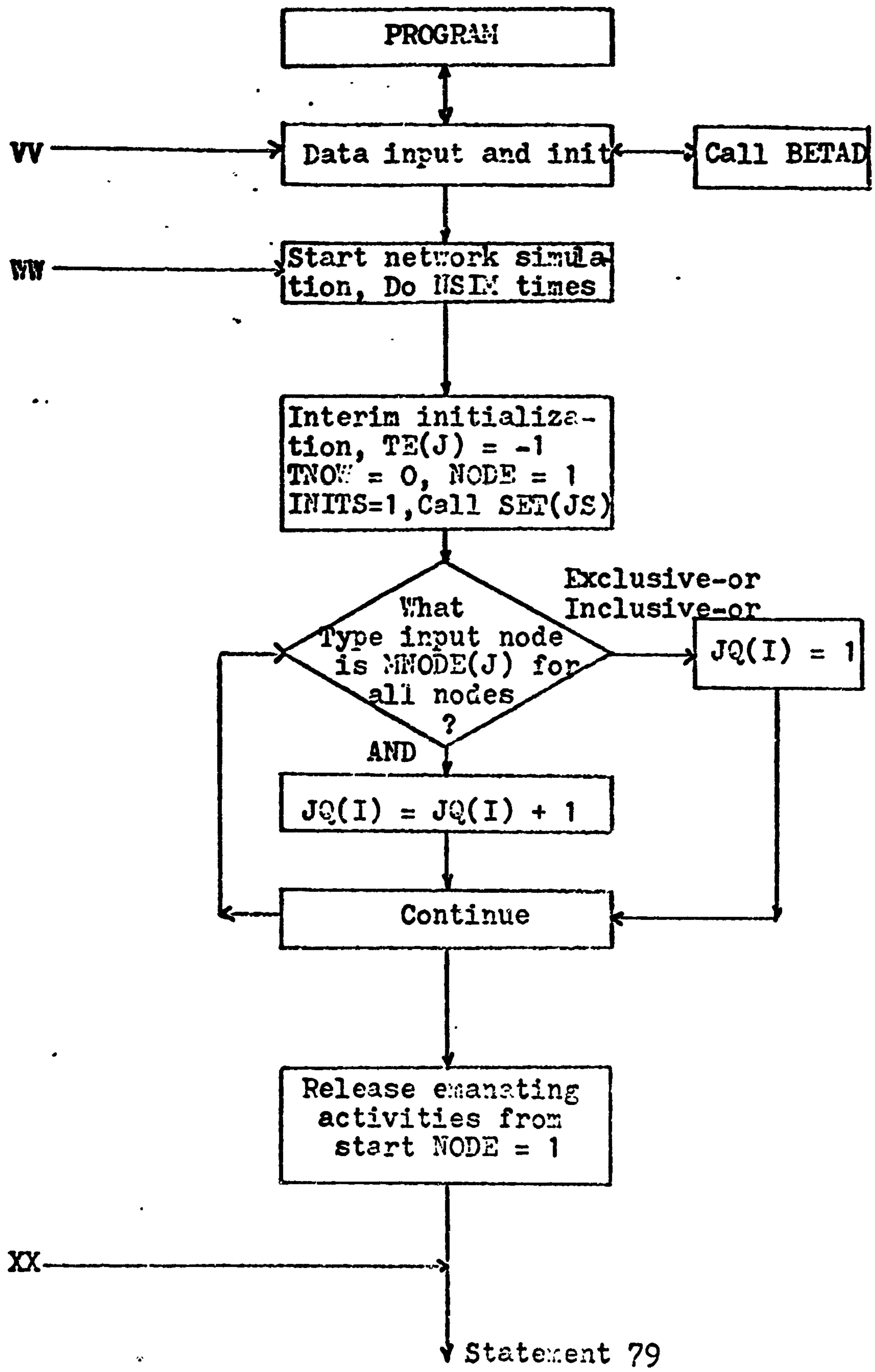


FIGURE 13

FLOW DIAGRAM OF THE MAIN PROGRAM

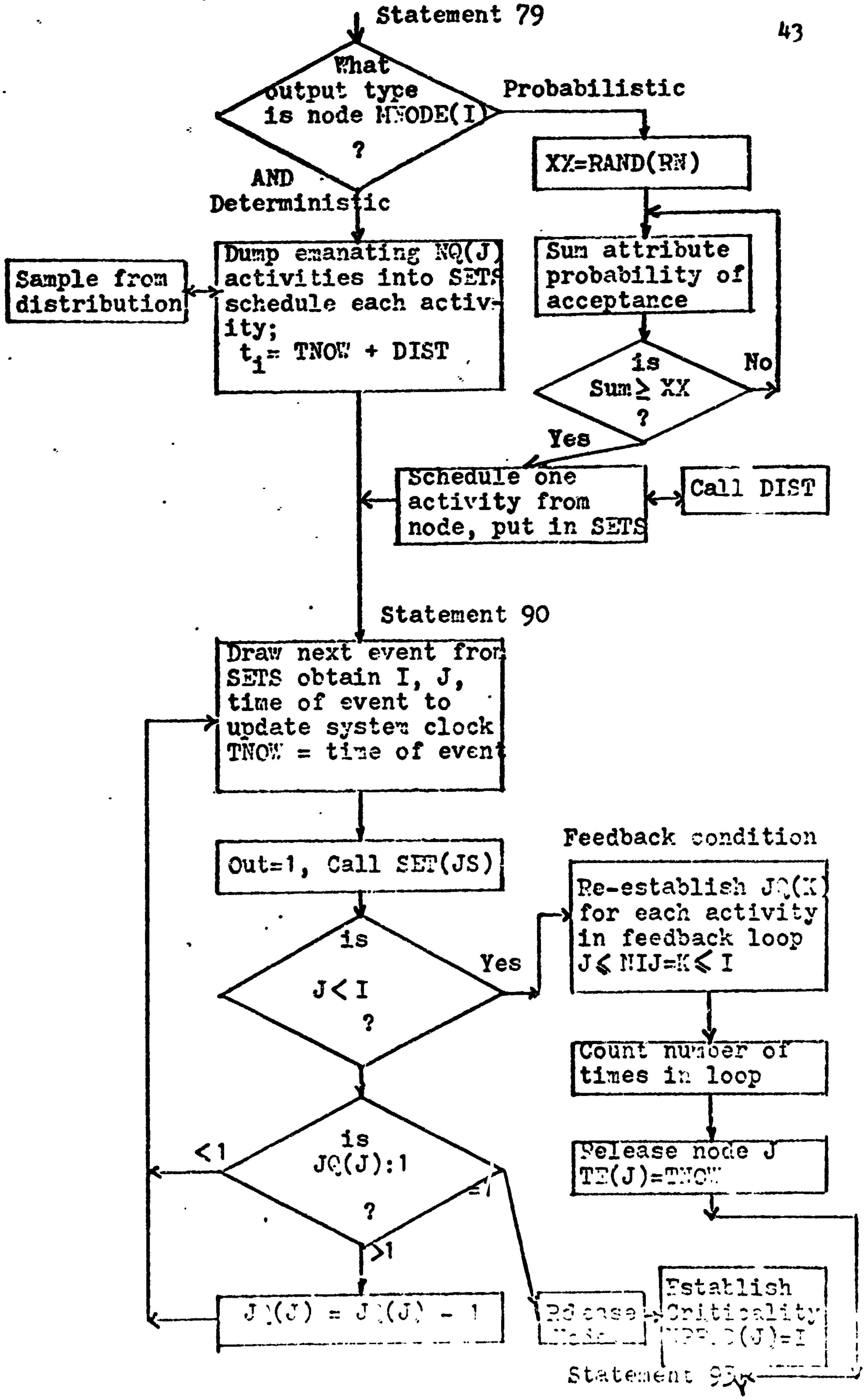


FIGURE 14. FLOW DIAGRAM OF THE MAIN PROGRAM

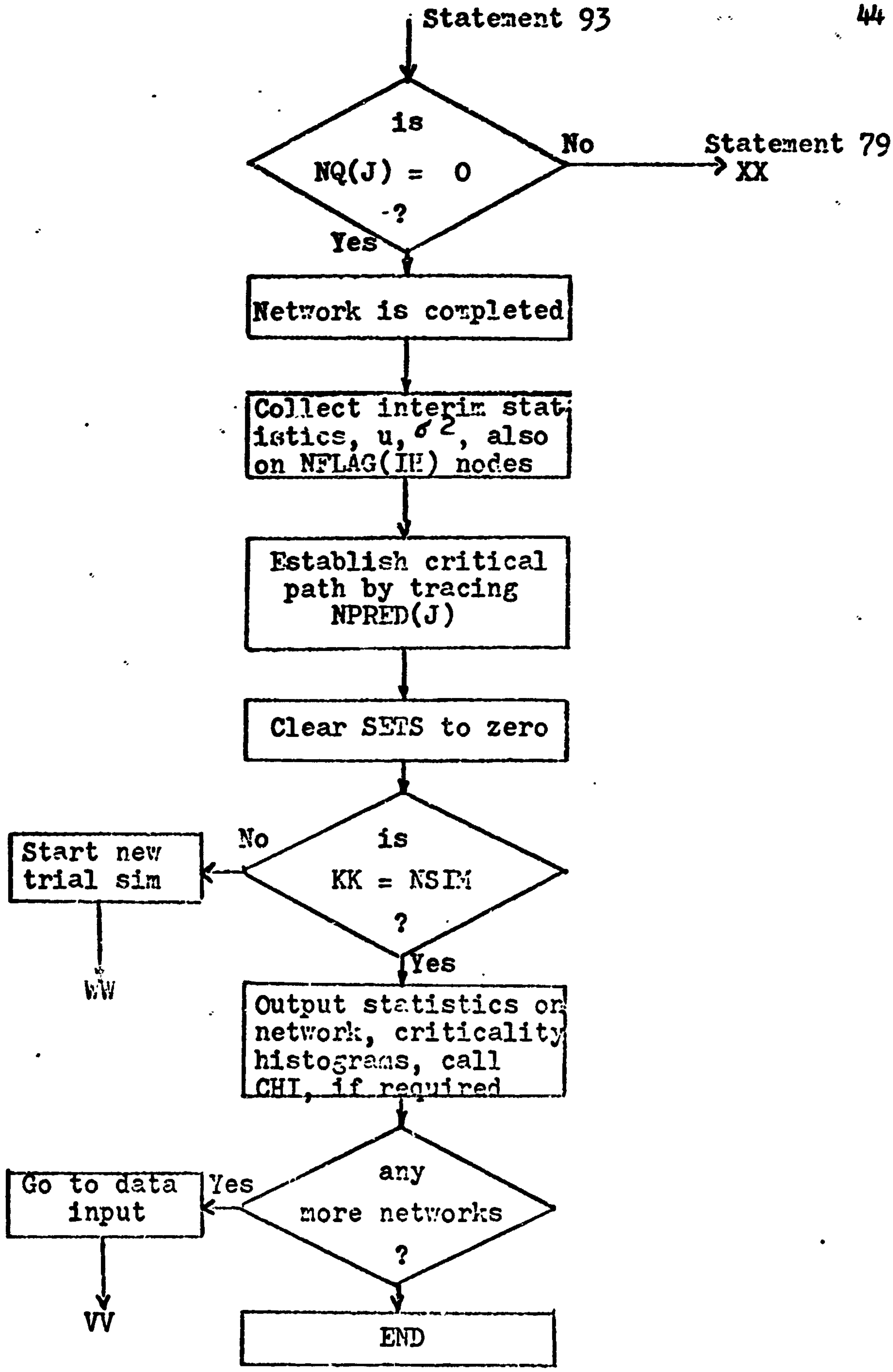


FIGURE 15

FLOW DIAGRAM OF THE MAIN PROGRAM

```
      PROGRAM MAC
      COMMON RN
      COMMON STATS(1000,4)
      COMMON SETS(40,5),INITS,OUT,KRANK(1),INN(1),MLC(1),MFA
      DIMENSION TE(400),MFLAG(4),NHST(4)
      DIMENSION A(8),X(400,8),NQ(100),JQ(100)
      DIMENSION NPRED(100),MNODE(100)
      DIMENSION NCRIT(100),STORE(400,8)
C
C     INITIALIZE ALL ARRAYS AND SUMS
C
    1 CONTINUE
      PRINT 114
      PRINT 120
      RN=.972
      CALL RAND(RN)
      VTIME=0.0
      ETIME=0.0
      JMAX=0
      N=0
      DO 296 I=1,400
      DO 296 J=1,8
  296 STORE(I,J)=0.0
      DO 299 I=1,60
      TE(I)=-1.0
      JQ(I)=0
      NQ(I)=0
      MNODE(I)=0
      NCRIT(I)=0
      DO 299 J=1,8
  299 X(I,J)=0.0
C
C     READING IN DATA IN INITIALIZATION
C
C     RN IS RANDOM NUMBER SEED AND NSIM IS NUMBER OF TRIAL SIMULAT
C
      READ 101,RN,NSIM
      IF(RN)888,888,3
    3 PRINT 101,RN,NSIM
C PUT DATA ON TAPE
      DO 50 M=1,400
      READ 126,(A(I),I=1,7)
      IF(A(1)-99.)49,51,51
   49 N=N+1
      A(8)=A(7)
      L=A(5)
      GO TO(45,45,4,45,44,45,45,45),L
    4 SIG=SQRTF(A(4))
      A(4)=A(3)-1.732051*SIG
      A(3)=A(3)+1.732052*SIG
      IF(A(4))951,45,45
  951 A(4)=0
      GO TO 45
   44 R=(A(4)-A(3))/(A(6)-A(3))
      CALL BETAD(R,ALPHA,GAMMA)
      A(7)=A(5)
```

```
      A(6)=A(3)
      A(3)=ALPHA
      A(4)=GAMMA
      GO TO 46
   45 CONTINUE
      A(6)=1.
      A(7)=1.
   46 NO(A(1))=NO(A(1))+1
      DO 555 I=1,8
  555 STORE(M,I)=A(I)
      JMAX=MAX1F(JMAX,A(2))
   50 CONTINUE
   51 CONTINUE
C
C     READ IN NODE INFORMATION
C     CODE FOR NODES EXCL-OR DET= 1 , EXCL-OR PROB= 2 , INCL-OR DET=
C     INCL-OR PROB= 4 , AND-DET= 5 , AND-PROB= 6 .
C
      PRINT 130
      PRINT 131
      DO 65 M=1,N
      READ 127,(NCRIT(II),II=1,24)
      IF(NCRIT(1)-99)62,66,62
   62 PRINT 128,(NCRIT(I),I=1,23,2)
      PRINT 129,(NCRIT(I),I=2,24,2)
      PRINT 131
      DO 63 NII=1,24,2
   63 MNODE(NCRIT(NII))=NCRIT(NII+1)
   65 CONTINUE
      IF(N-30)66,66,61
   61 PRINT 114
      PRINT 131
   66 DO 67 L=1,60
   67 NCRIT(L)=0
      DO 280 L=1,4
      NFLAG(L)=0
  280 NHST(L)=0
C
C     NFLAG DENOTES THE NODE BEING ANALYSED, MAXIMUM OF 4 NODES
      READ 105,(NFLAG(I),I=1,4)
C
C INITIALIZE SETS
C
      JS=1
      INN(JS)=1
      INITS=1
      CALL SET(JS)
      NX=1
      JS=1
      KRANK(JS)=1
C TAKE DATA FROM TAPE IN ORDER BY I NODE
      DO 69 M=1,N
      DO 600 K=1,8
  600 A(K)=STORE(M,K)
      NAA=A(1)
      NCOUNT =0
```

```
      IF(NAA-1)100,59,52
   52 NA=NAA-1
      DO 58 K=1,NA
   58 NCOUNT=NCOUNT+ND(K)
   59 NA=NCOUNT+1
   53 IF(X(NA,1))100,56,55
   55 NA=NA+1
      GO TO 53
   56 DO 57 L=1,8
   57 X(NA,L)=A(L)
   60 CONTINUE
C CHECK BUFFER AND ND(I) FROM MAIN
      PRINT 103,((X(M,K),K=1,8),M=1,N)
      PRINT 106,N
      IF(N-15)915,915,916
  916 PRINT 114
      PRINT 151
  915 CONTINUE
C
C START TRIAL SIMULATION
      DO 995 KK=1,NSIM
C INITIALIZE TRIAL SIMULATION
      DO 991 I=1,60
      TE(I)=-1.0
  991 JQ(I)=0
      DO 75 K=1,N
      L=X(K,2)
      LLL=MNODE(L)
      GO TO (11,11,11,11,12,12),LLL
   11 JQ(L)=1
      GO TO 75
   12 JQ(L)=JQ(L)+1
   75 CONTINUE
      KRANK(1)=3
      TNOW=0.
      NODE=1
      I=1
      J=1
C
C RELEASE *NODE* SCHEDULE ALL EMANATING ACTIVITIES
C
   79 XNODE=NODE
      DO 155 L=1,N
C
C DUMP EMANATING ACTIVITIES INTO SETS
C
  152 IF(X(L,1)-XNODE)155,5,155
    5 NN=ND(NODE)+L-1
      LLL=MNODE(J)
      GO TO (153,6,153,6,153,6),LLL
    6 XX=RAND(RN)
      SUM=0.
      DO 8 K=L,NN
      SUM=SUM+X(K,8)
      IF(SUM-XX)8,8,7
    7 SETS(IFA,1)=X(K,1)
```

```
      SETS(MFA,2)=X(K,2)
      SETS(MFA,3)=TNOW+DIST(X(K,3),X(K,4),X(K,5),X(K,6),X(K,7))
      CALL SET(JS)
      GO TO 90
    8 CONTINUE
  153 CONTINUE
      DO 154 K=L,NN
      SETS(MFA,1)=X(K,1)
      SETS(MFA,2)=X(K,2)
      SETS(MFA,3)=TNOW+DIST(X(K,3),X(K,4),X(K,5),X(K,6),X(K,7))
      CALL SET (JS)
  154 CONTINUE
      GO TO 90
  155 CONTINUE
C
C UPDATE CLOCK AND DRAW NEXT EVENT
C
   90 TNOW=SETS(MLC(1),3)
      J=SETS(MLC(1),2)
      I=SETS(MLC(1),1)
      OUT=1.
      CALL SET(JS)
      IF(J-I)95,100,192
  192 IF(JQ(J)-1)91,92,93
   95 DO 98 IJ=1,N
      NIJ=X(IJ,1)
      IF(NIJ-J)98,97,96
   96 IF(NIJ-I)97,94,91
   94 NCRIT(IJ)=NCRIT(IJ)+1
   97 LLL=MNODE(NIJ)
      GO TO (15,15,15,15,16,16),LLL
   15 JQ(NIJ)=1
      GO TO 98
   16 JQ(NIJ)=JQ(NIJ)+1
   98 CONTINUE
      GO TO 91
   92 NPRED(J)=I
   91 JQ(J)=0
      NODE=J
      TE(J)=TNOW
      IF(NQ(J))100,320,79
   93 JQ(J)=JQ(J)-1
      GO TO 90
  100 PRINT 199
  199 FORMAT(5HERROR)
      PRINT 106,I,J
      GO TO 1
  320 CONTINUE
      ETIME=ETIME+TE(J)
      VTIME=VTIME+TE(J)*TE(J)
      DO 222 IH=1,N
      IF(NFLAG(IH))222,222,226
  226 IF(TE(NFLAG(IH)))222,222,229
  229 NT=IH+1
      NHST(IH)=NHST(IH)+1
      STATS(NHST(IH),IH)=TE(NFLAG(IH))
```

```
  222 CONTINUE
C
C ESTABLISH CRITICAL EVENT
C
      MN=J
      NN=N
  201 L=MPRED(MN)
      XL=L
      DO 205 K=1,NN
      NROW=NN-K+1
      IF(X(NROW,1)-XL)205,202,205
  202 IF(X(NROW,2)-MN)205,203,205
  203 NCRIT(NROW)=NCRIT(NROW)+1
  205 CONTINUE
      MN=L
      IF(L-1)206,206,201
  206 CONTINUE
      IF(MLC(JS))995,995,905
  905 INITS=1
      CALL SET (JS)
  995 CONTINUE
C
C     CALCULATE STATISTICS
C
      XNSIM=NSIM
      ETIME=ETIME/XNSIM
      VTIME=VTIME/XNSIM-ETIME*ETIME
      PRINT 108,ETIME
      PRINT 109,VTIME
C
C     CRITICALITY CALCULATIONS
      PRINT 110
      DO 200 I=1,N
      CRIT=NCRIT(I)
      B=CRIT/XNSIM
      PRINT 111,X(I,1),X(I,2),B
  200 CONTINUE
      PRINT 114
      NN=0
C
C     HISTOGRAM CALCULATIONS
      DO 422 IH=1,4
      IF(NFLAG(IH))422,422,420
  420 NT=IH
      PRINT 131
      PRINT 112,NFLAG(IH)
      PRINT 131
      PRINT 132,NHST(IH)
      XH=NHST(IH)
      TT=XH/XNSIM
      PRINT 133,TT
      CALL HIST(NT,NHST(IH))
      IF(IH-2)422,421,422
  421 PRINT 114
  422 CONTINUE
      GO TO 1
```

```
 887 CONTINUE
 888 CALL EXIT
 101 FORMAT(F5.3,I4)
 102 FORMAT(4F3.0,F5.0)
 103 FORMAT(5F10.2)
 106 FORMAT(20I4)
 108 FORMAT(26X,27HEXPECTED TIME LAST EVENT IS,F10.4/)
 109 FORMAT(26X,27HVARIANCE TIME LAST EVENT IS,F10.4/)
 110 FORMAT(/31X,17HCRITICALITY INDEX,/30X,4H   I,4H   J,10H       CR
    1)
 111 FORMAT(30X,2F4.0,F10.2)
 112 FORMAT(27X,21H*HISTOGRAM DATA, NODE,I4,1H*,/)
 114 FORMAT(1H1)
 115 FORMAT(/20I4)
 120 FORMAT (/27X,25H*GERT NETWORK SIMULATION*//)
 126 FORMAT(2F5.0,2F10.2,3F5.0)
 127 FORMAT(24I3)
 128 FORMAT(18X,5H*NODE,2X,12I3,1H*)
 129 FORMAT(18X,5H*TYPE,2X,12I3,1H*)
 130 FORMAT(34X,12H*INPUT DATA*)
 131 FORMAT(//)
 132 FORMAT(20X,27HNUMBER OF NODE REALIZATIONS,I10)
 133 FORMAT(/,20X,29HPROBABILITY OF REALIZATION IS,F10.4/)
 999 FORMAT(5F10.4)
     END
```

```
      FUNCTION RAND(X)
      X=X*1.0E+8*23
      I=X*1.0E-8
      Y=I
      X=(X-Y-Y*1.0E+8)*1.0E-8
      RAND=X
      RETURN
      END
```

CRIT/

A maximum of four nodes may be flagged for analysis during the simulation of the network by including them on the next data input card. These node numbers are retained as values of the variable NFLAG(IH).

A buffer array, X(M, K), is utilized as the permanent storage area of the activities and their attributes, which are transformed and ordered by I node designations from the array STORE (N, I). The contents of X(M, K) are then printed as output.

The basic procedure in simulating the network is one of working through the network from node to node, releasing activities and scheduling them in the master time file SETS(J, K), and recalling the activities from SETS(J, K) as they have been completed.

The actual simulation of a network can be more easily explained by tracing through a simulation cycle.

Each cycle begins with initialization of TE(I), time node I is released; TNOW, current time of system; and JQ(J), required number of activities terminating at a node necessary for realization of the node. JQ(J), for all nodes, is set equal to 1 for Exclusive-or and Inclusive-or type nodes, and $>$ 1 for AND type nodes. The start node is then realized by setting NODE = 1, the first node of the network.

As each NODE is realized, control is transferred to statement 79, which causes the emanating activities to be released and scheduled. Should the output node types be deterministic, all activities emanating from that node are scheduled and placed in the master time file SETS(I, K) with scheduled time equal to TNOW + DIST. As previously mentioned, the variable DIST takes on a time value as a random variable of the probability function for each activity released.

Should the output node type be probabilistic, a random number is generated and used as a reference in determining which activity of the activities emanating from the node should be scheduled. Only one activity is scheduled with time equal to TNOW + DIST; the others are dropped.

Each act of placing an activity in SETS is accompanied by a call of Subroutine SET(1), which causes ordering of the activities in sets through a marker arrangement as previously mentioned.

After the appropriate number of activities have been released and filed in SETS(I, K), control is advanced to statement 90, which causes the next event to occur. The activity filed in SETS(I, K) marked for next exit by MLC(1) is drawn from SETS(I, K) and the system time, TNOW, is updated. TNOW is set equal to the scheduled time of completion of that activity.

A comparison of the I and J attributes of the completed activity is made to see if a feedback condition exists, where $J < I$. If not, then JQ(J), number of completed activities required node J, is investigated. If $JQ(J) > 1$, then $JQ(J) = JQ(J)-1$, which means more activities have yet to be completed before node J can be realized. Control is returned to statement 90 for next release of a scheduled activity from SETS(I, K).

If $JQ(J) = 1$, node J is released and the time of release, TE(J), is set to TNOW. Precedence for criticality measurement is established by setting $NPRED(J) = I$.

The variable NQ(J), or number of activities emanating from node J, is investigated to determine if an exit node has been realized. If $NQ(J) > 0$, control is transferred to statement 79 with NODE = J for release of node J as described above. When $NQ(J) = 0$, the network has been completed since J was a terminating node, and statistics are compiled.

Control is returned to the beginning of the simulation phase for another network simulation. This process continues until the required number of simulation trials, NSIM, has been attained. Final output statistics are then computed and printed.

Intentionally omitted from the preceding discussion is the special case of feedback, where $J < I$ for a completed activity. Feedback returns network control to a node previously realized in the network. A meaningless situation could exist since scheduled activities positioned between nodes I and J may not have been completed and are still in the master time file. The network of nodes and activities comprising a feedback loop is depicted in Figure 16.

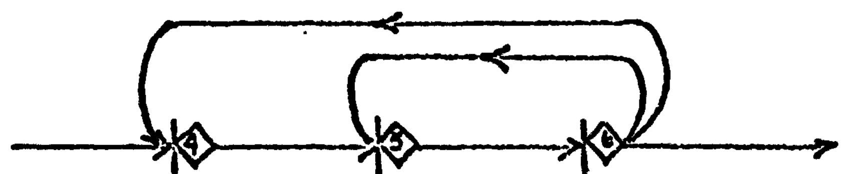


FIGURE 16
FEEDBACK NETWORK

Nodes 4, 5, and 6 define the feedback loop, and the node numbering is such that $J < NIJ < I$. NIJ represents the nodes positioned within the loop.

Another situation that could exist stems from the fact that each JQ(J) within the feedback loop has been set to zero as the nodes were released. This programming aspect is corrected by re-establishing JQ(J) for each node NIJ and I within the feedback loop, $J < NIJ < I$. Thus, recycling is permitted for this feedback portion of the network.

Caution is advised in designing the network model to eliminate undesirable feedback situations. For example, consider the network shown in Figure 17 consisting of one feedback loop with one activity emanating

from a node within the loop to the external network.

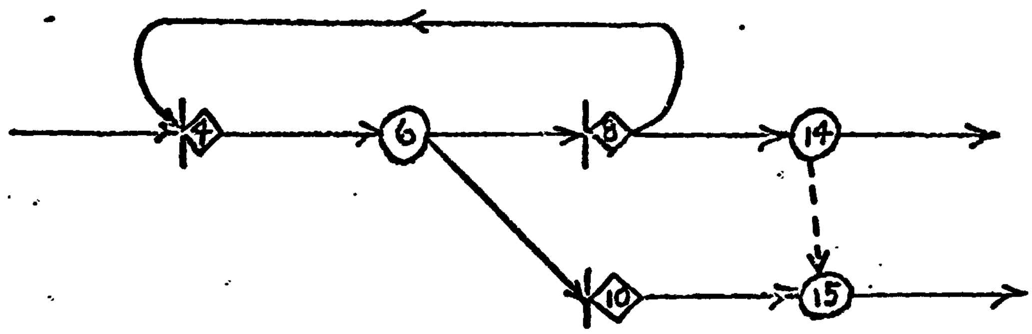


FIGURE 17
ONE LOOP NETWORK

As the network is constructed, the first realization of node 6 would release activities 6, 8 and 6, 10. The network beyond node 10 would then be activated. Additional realizations of node 10 by repeated looping among nodes 4, 6, 8, 4 could not cause another release of node 10 since JQ(10) = 0 after the first realization. However, should the analyst desire to halt realization of the external network beyond node 10 until activity 8, 14 is released, a restraining activity 14, 15 could be constructed. Node 10 can have only one terminating activity for this scheme to work properly.

Consider a more complex network than that discussed above where a second feedback loop is compounding the effects of the first. The network is shown in Figure 18.

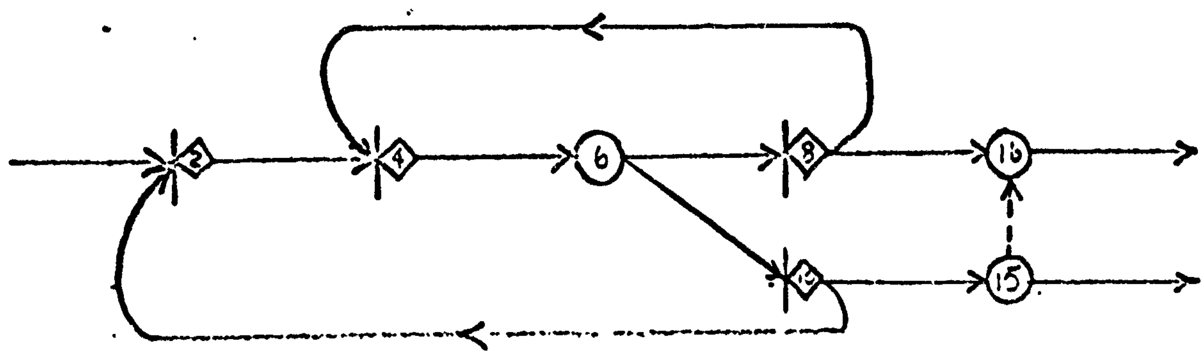


FIGURE 18
TWO LOOP NETWORK

Since JQ(K), for J < K < I is reset by the program in a feedback loop, repeated looping from nodes 4, 6, 8, 4 will not cause more than one release of node 10 due to this loop alone. However, should activity 10, 2 be released, feedback looping would occur in both loops. Restraining activity 15, 16, inserted by the analyst would prevent the possibility of repeated releases from node 8 to the external network, since each realization of the loop 10, 2, 4, 6, 10 would cause node 8 to be realized.

IV. INPUT-OUTPUT

An important attribute of the computer program is mimimization of user-time in preparation of input data for the simulation program. This section describes the input data cards for computer processing of input data and program output. A sample listing of input data and output is provided for reference on pages 57 through 60.

Only one input card per activity is required containing I-J node designations, parameters or time-estimates describing the probability density function, code for the density function, and probability of acceptance should the I node be realized.

Codes for the probability density functions and input parameters or time estimates are listed as follows.

Density Function	Code	Parameters or Estimates
Exponential	1	u
Normal	2	u σ^2
Rectangular Discrete	3	u σ^2
Constant	4	T
Beta	5	a m b

```
      1    2     4.00      1.00    4
      1    3     1.00      1.00    4
      2    3     3.00      1.00    4
      2    4     2.00      1.00    4
      2    5     5.00      1.00    4
      3    4     3.00      1.00    4
      4    5     1.00      1.00    4
      4    6     4.00      1.00    4
      5    6     2.00      1.00    4
   99
    1  5  2  5  3  5  4  5  5  5  6  5  7  5  8  5  9  5 10  5
   99
     6
  004770100
      1    2     4.00      1.00    2
      1    3     1.00      1.00    2
      2    3     3.00      1.00    2
      2    4     2.00      1.00    2
      2    5     5.00      1.00    2
      3    4     3.00      1.00    2
      4    5     1.00      1.00    2
      4    6     4.00      1.00    2
      5    6     2.00      1.00    2
   99
    1  5  2  5  3  5  4  5  5  5  6  5  7  5  8  5  9  5 10  5
   99
     4    6
  004771000
      1    2                       4         1.0
      2    3       2.              4          .2
      2    5       2.              4          .5
      2    7       2.              4          .3
      3    4                       4          .1
      3    5       2.              4          .3
      3    6                       4          .3
      3    7       2.              4          .3
      4    3       1.              4         1.0
   99
    1  5  2  2  3  2  4  5  5  2  6  2  7  2
   99
     4    5    6    7
```

SAMPLE INPUT DATA

GERT NETWORK SIMULATION

.4771000

INPUT DATA

*NODE	1	2	3	4	5	6	7	-0	-0	-0	-0	-0*
*TYPE	5	2	2	5	2	2	2	-0	-0	-0	-0	-0*

1.00	2.00	-0.00	-0.00	4.00	1.00	1.00	1.00
2.00	3.00	2.00	-0.00	4.00	1.00	1.00	0.20
2.00	5.00	2.00	-0.00	4.00	1.00	1.00	0.50
2.00	7.00	2.00	-0.00	4.00	1.00	1.00	0.30
3.00	4.00	-0.00	-0.00	4.00	1.00	1.00	0.10
3.00	5.00	2.00	-0.00	4.00	1.00	1.00	0.30
3.00	6.00	-0.00	-0.00	4.00	1.00	1.00	0.30
3.00	7.00	2.00	-0.00	4.00	1.00	1.00	0.30
4.00	3.00	1.00	-0.00	4.00	1.00	1.00	1.00

9

EXPECTED TIME LAST EVENT IS 2.2930

VARIANCE TIME LAST EVENT IS 0.5972

CRITICALITY INDEX

I	J	CRIT
1	2	1.00
2	3	0.21
2	5	0.48
2	7	0.31
3	4	0.00
3	5	0.08
3	6	0.08
3	7	0.05
4	3	0.04

HISTOGRAM DATA, NODE 4

NUMBER OF NODE REALIZATIONS 29

PROBABILITY OF REALIZATION IS 0.0290

EXPECTED TIME OF COMPLETION THIS NODE IS 2.2759

VARIANCE TIME OF COMPLETION THIS NODE IS 0.4067

2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	2
23	5	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

BIG	SMALL	RANGE
5.000	2.000	3.000

HISTOGRAM DATA, NODE 5

NUMBER OF NODE REALIZATIONS 563

PROBABILITY OF REALIZATION IS 0.5630

EXPECTED TIME OF COMPLETION THIS NODE IS 2.3321

VARIANCE TIME OF COMPLETION THIS NODE IS 0.7014

2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	2
80	0	69	9	4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	

BIG	SMALL	RANGE
8.000	2.000	6.000

HISTOGRAM DATA, NODE 6

NUMBER OF NODE REALIZATIONS 80

PROBABILITY OF REALIZATION IS 0.0800

EXPECTED TIME OF COMPLETION THIS NODE IS 2.1125

VARIANCE TIME OF COMPLETION THIS NODE IS 0.0998

2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	2
71	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

BIG	SMALL	RANGE
3.000	2.000	1.000

HISTOGRAM DATA, NODE 7

NUMBER OF NODE REALIZATIONS 357

PROBABILITY OF REALIZATION IS 0.3570

EXPECTED TIME OF COMPLETION THIS NODE IS 2.2717

VARIANCE TIME OF COMPLETION THIS NODE IS 0.5340

2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	2
312	0	39	5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

BIG	SMALL	RANGE
6.000	2.000	4.000

Time estimates a, m, and b for the Beta Distribution are converted at input time through SUBROUTINE BETAD into parameters α and γ of the Beta probability density function.

The data input cards for each activity are punched in FORMAT (2I5, 2F10.2, 3F5.0), in the following sequence for distribution codes 1-4,

I J u σ^2 Code

and in the following sequence for distribution code 5

I J a m Code b

GERT logic node types and codes are given below.

INPUT	OUTPUT	CODE
Inclusive-or	deterministic	1
	probabilistic	2
Exclusive-or	deterministic	3
	probabilistic	4
AND	deterministic	5
	probabilistic	6

Each network node from node 1 to the end nodes is defined with its respective node type by input to the program in FORMAT (24I3) as indicated below.

Node	Type	Node	Type	Node	Type
1	5	2	5	3	4

As many data cards are utilized as necessary to input the nodes and types.

Nodes that are to be analyzed, causing a complete print out of statistics gathered, are denoted as NFLAG and entered next in FORMAT (4I4).

The data deck, along with the necessary control cards are listed below:

Card Column	
1 2 3 4 5 6 7 8 9 10 11 12	
0 0 4 7 7 1 0 0 0	Random number seed and NSIM
I J	Activity cards, one per activity
0 9 9	End of activity list
0 0 1 0 0 5 0 0 2 0 0 5	Node numbers and types, 12 sets per data card
0 9 9	End of node list
0 0 0 6 0 0 0 7	Nodes to be analyzed

As many networks may be analyzed as desired in one computer run by providing the sequence of data cards described above for each network. The network sets of data cards are stacked one on the other with a blank card placed at the end of the last network to signify end of program data.

Should the Chi Square Goodness of Fit Test be required; ten input cards, each containing the expected cell block size obtained from the tabulated probability distribution function utilized, is entered in FORMAT (F6.4). The computer program is designed for symmetrical distributions, so the ten equally spaced cells would represent the area under the density function from the mean to one tail.

Computer output is divided into five sections: (1) input data, (2) expected and variance of network completion time, (3) criticality

index, (4) node statistics, and if required (5) chi square test results. Discussion follows based on this division.

An echo check of the random number seed, and number of simulation trials precedes a complete print out of the activities and their respective attributes. Each activity contained in the buffer storage array is printed in the form utilized during simulation, as follows for distribution codes 1-4.

I J u σ^2 code - - p_i

For activities distribution code 5, the following is printed out.

I J α γ code a b p_i

The next section provides the expected and variance of network completion time regardless of the number of end nodes.

A criticality index follows for all activities in the network as the relative frequency of occurrence of an activity on the critical path of the network. Activities possessing the I > J feedback characteristic will have recorded, not criticality, but total number of times realized divided by number of simulation trials since many realizations of a loop may occur during each simulation trial.

For each node flagged at input for analysis, the output will consist of, number of realizations, probability of realization, mean and variance of realization time, histogram of these times, high time, low time, and range.

The Chi Square Goodness of Fit Test is printed out only when Subroutine CHI is utilized and theoretical distribution input provided. The calculated chi square value, mean and variance of the sample distribution, and histogram of the normalized distribution is provided.

CHAPTER IV

APPLICATIONS OF THE SIMULATION MODEL

The purpose of this chapter is to present several applications of GERT network analysis through use of the GERT Simulation Program. Several example problems have been chosen to demonstrate general applicability of the computer program to various type networks.

Three sample probabilistic networks are simulated and the results compared with analytical solutions such as from the GERT computer program described by Pritsker.[1] The aim is not to perform a detailed analysis of the individual networks; moreover, to indicate the approach to problem solution with the GERT Simulation Program and to provide a measure of the validity of simulated results.

One other network, consisting of AND nodes, is considered to investigate assumptions developed in the literature on PERT networks. The resultant distributions are analyzed using normally distributed and beta distributed activity times. The Chi Square Goodness of Fit Test is applied to test the hypothesis of normally distributed network realization times.

I. PROBABILISTIC NETWORKS

The first network depicts an analysis of a Research and Develop-

[1]A. Alan B. Pritsker, GERT: Graphical Evaluation and Review Technique, The RAND Corporation, RM-4973-NASA (Santa Monica, April, 1966), Appendix A.

ment project given in a recent article by Graham,[2] and discussed in relation to GERT, by Pritsker.[3] The network is constructed in Figure 19 For each branch or activity of the network, the probability that the branch is realized given the preceding node is realized is depicted by p_i, and the constant time associated with activity k is depicted by t_k. Note that there are two terminating nodes of this network; success, s; and failure, f.

The criticality index of the relative frequency an activity contributed to network realization, is presented in Table I. The ability of the simulation model to select activities emanating from a probabilistic output node can be checked by comparing the probabilities associated with activities 1-2 and 1-6, also activities 6-9 and 6-8, in Figure 19 with the criticality index of the activities in Table I. Activities 1-2 and 1-6 were defined with probabilities of 0.7 and 0.3 which compare favorable with simulated criticality of 0.68 and 0.32, respectively. Similarily, activities 6-8 and 6-9 defined with probabilities of (.3)(.5) each compare with simulated criticality of 0.17 and 0.16. The slight difference in each case can easily be attributed to sampling error.

---

[2] Pearson Graham, "Profit Probability Analysis of Research and Development Expenditures", The Journal of Industrial Engineering, Vol. XVI, No. 3, May June 1965, pp. 186-191.

[3] Pritsker, Op. Cit., p. 57.

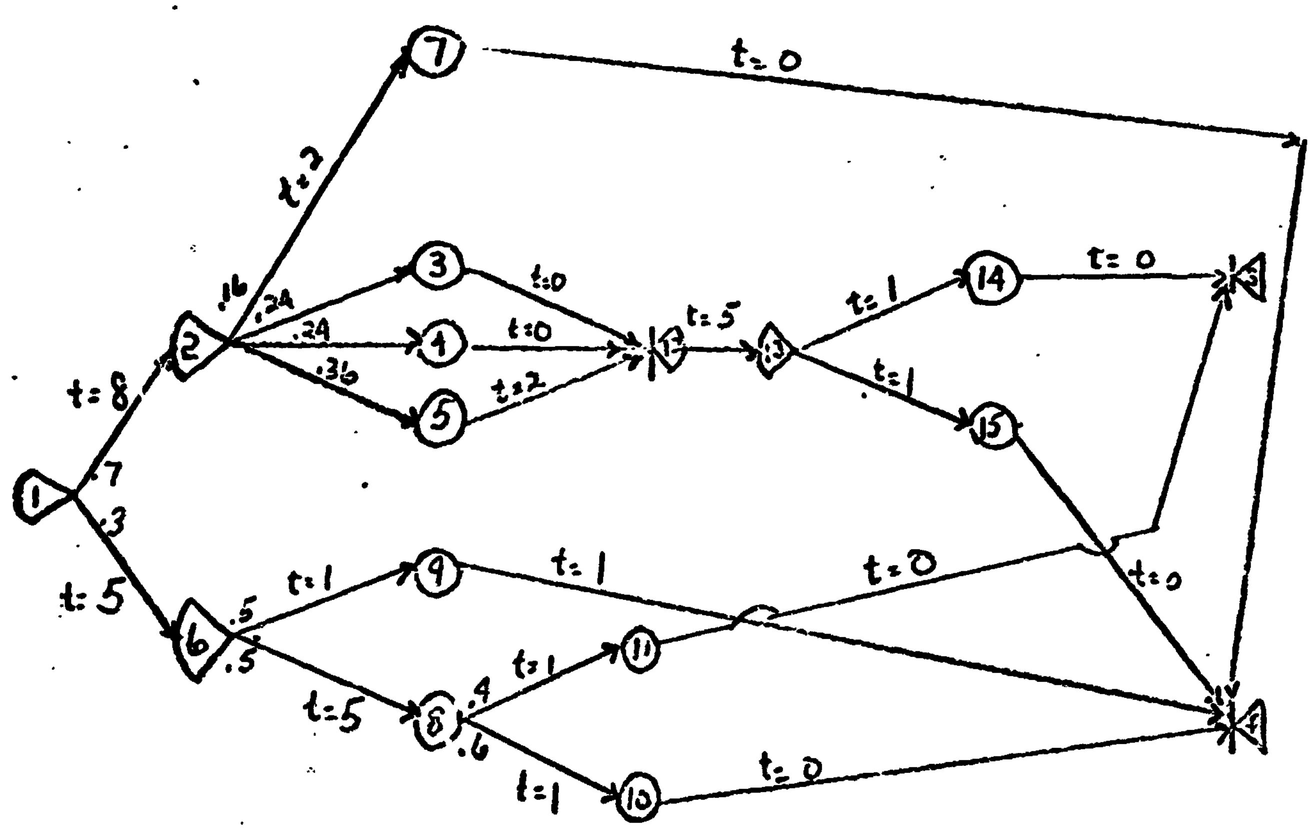


FIGURE 19

GERT NETWORK FOR RESEARCH AND DEVELOPMENT PROBLEM

TABLE I

SIMULATED CRITICALITY INDEX ON ACTIVITIES IN THE RESEARCH AND DEVELOPMENT PROBLEM

Activity	I	J	CRITICALITY
	1	2	0.68
	1	6	0.32
	2	3	0.18
	2	4	0.15
	2	5	0.24
	2	7	0.10
	3	12	0.18
	4	12	0.15
	5	12	0.24
	6	8	0.17
	6	9	0.16
	7	17	0.10
	8	10	0.09
	8	11	0.07
	9	17	0.16
	10	17	0.09
	11	16	0.07
	12	13	0.57
	13	14	0.40
	13	15	0.17
	14	16	0.40
	15	17	0.17

Node analysis was conducted on nodes 16 and 17 for probability of realization and resultant distribution of realization times. A test of proportions using the normal approximation is utilized to test the hypothesis that the simulated probability of realization is equal to the analytical probability, at the 5 per cent level of significance, for each possible time of realization.

The test statistic is, $Z = (X - Np)/\sqrt{Npq}$, where Z is the standardized variable, X is the simulated frequency of successes, p is the analytic proportion of successes, q is the proportion of failures,

$q=1-p$, and N is the sample size. A two sided test is applied with a critical region, $-1.96 < Z < 1.96$. Results of the node analysis are as depicted in Table II where the Z score is given for each pair of simulated and analytical frequencies at the discrete time intervals.

TABLE II

RESULTS OF NODE ANALYSIS ON RESEARCH AND DEVELOPMENT PROBLEM

Node	Time of Realization	Simulation Frequency	Analytical Frequency	Z Statistic
16	11.0	74.0	60.0	.904
	16.0	398.0	411.6	-.875
17	6.0	158.0	150.0	.710
	10.0	104.0	112.0	-.805
	11.0	91.0	90.0	.111
	16.0	175.0	176.0	.116

The test of proportions utilized indicates there is no reason to reject the hypothesis that the simulated and analytic results are equal, as all Z scores are well within the critical region.

The second network considered for simulation is the Thief of Bagdad abstracted from Parzen[4] and analyzed in GERT[5]. The problem concerns a thief in a dungeon faced with the selection of three doors

[4]Emanuel Parzen, Stochastic Processes, (San Francisco: Holden-Day, Inc., 1962), p. 50.

[5]Pritsker, op. cit., p. 45.

leading to; (1) freedom; (2) a long tunnel returning to the dungeon; and (3) a short tunnel returning to the dungeon. Experience has no effect on selection given he makes an incorrect choice and returns to the dungeon. The network is depicted in Figure 20 with constant activity time assignments, t, and probability of selection, p.

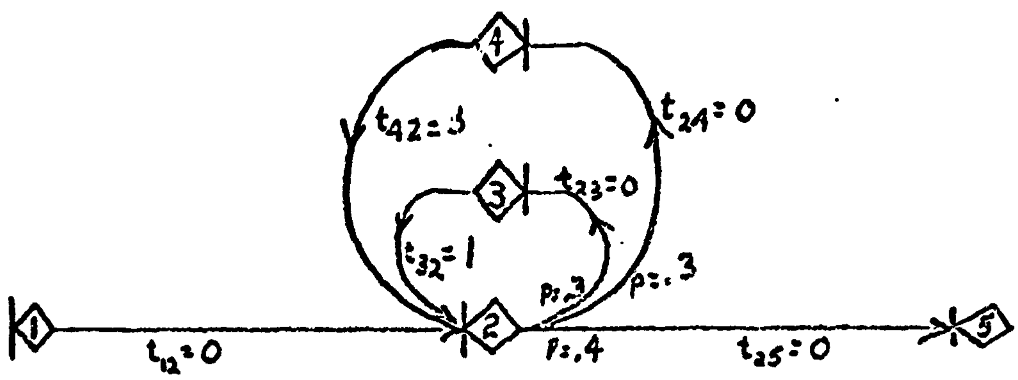


FIGURE 20

GERT NETWORK FOR THE THIEF OF BAGDAD PROBLEM

This network was chosen because the feedback loops, one parallel to the other, would provide a challenge to the computer program.

The expected value of node 5, u_5, for the 1000 simulated trials was 3.057 with variance equal to 17.684. The criticality index of .76 for activity 3-2 and .77 for activity 4-2 indicates equal probability of selection. The histogram of realization times for node 5 is shown in Figure 21.

The expected time to realize node 5 can be determined by,[6]

$$E(t_5) = t_{12} + t_{25} + 1/p_{25}(\, p_{24}t_{24} + p_{23}t_{23}\,)$$

$$= 0 + 0 + .4(\, .9 + .3\,) = 3.0$$

[6] Ibid.

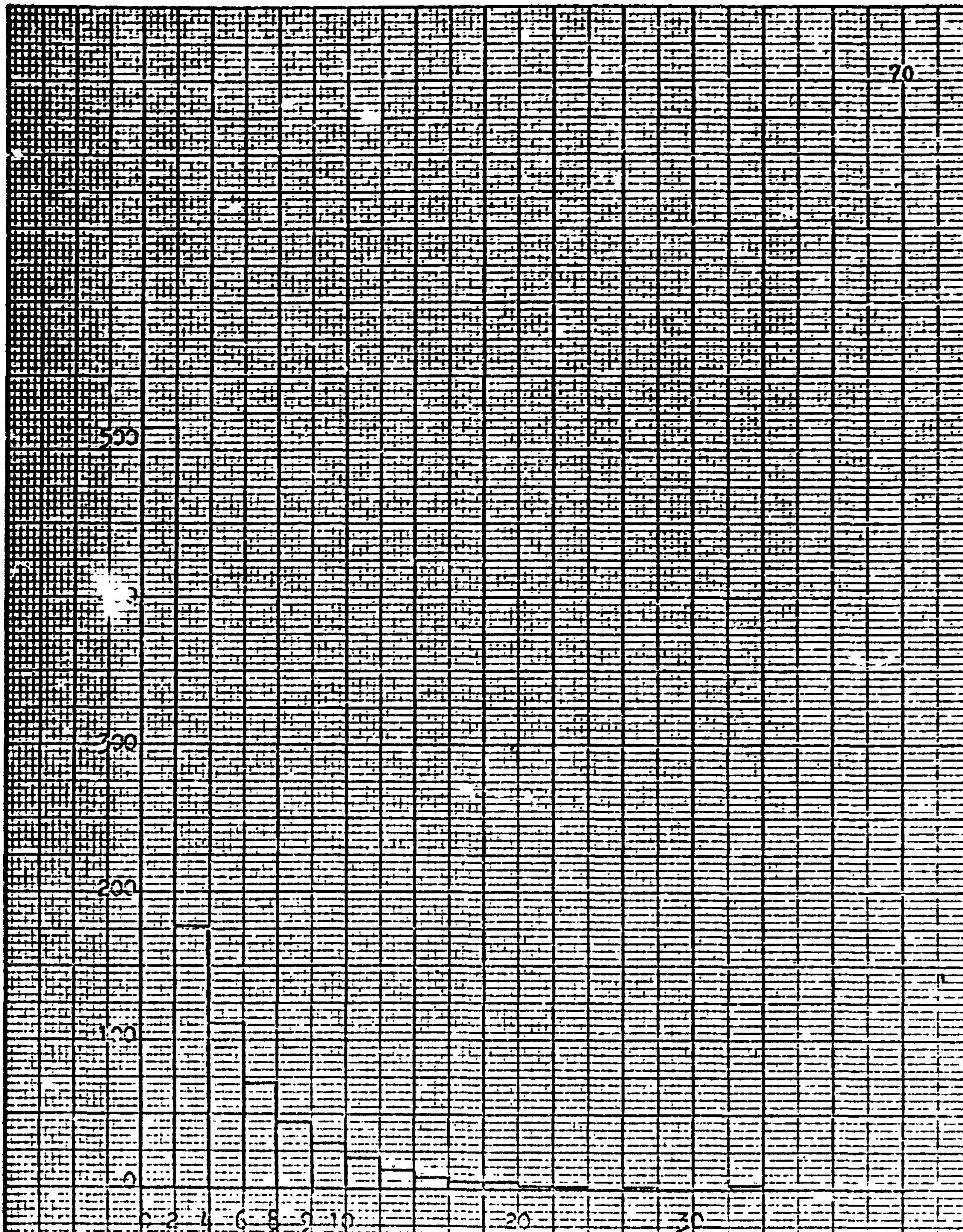


FIGURE 21

HISTOGRAM OF MODE 5 REALIZATION TIMES

A favorable comparison is noted between analytical and simulated values for expected time of realization, node 5, of 3.0 and 3.057.

A test of proportions is utilized to further check the accuracy of the simulation results for probability of realizing node 5 in less than 2 time units is

$$p = p(t_5 < 2) = p(t=0) + p(t=1) = .4 + (.4)(.3) = .52$$

As defined for the previous network, the test statistic is $Z = (X - Np)/\sqrt{Npq}$. Application of the test to the simulated results of .513 and analytical results of .52 for $p(t_5 < 2)$ at the 5 per cent level of significance results in the conclusion of no difference in the two procedures for this time interval.

The bomber-interceptor problem is chosen as the third example of networks containing probabilistic type nodes. An interceptor may kill or be lled on the first pass at the bomber or it may miss and make another pass at the bomber, again facing the chance of killing, be killed, or missing. Successive passes are allowed with the chance of mission abort after each pass. The network is depicted in Figure 22 with probabilities, p, of activity realization and constant times associated with each activity. Note that a small feedback loop represents the successive passes of the interceptor.

The criticality index for each activity in the network is presented in Table III. Comparison with the probabilities as given in the network indicate only slight differences attributed to sampling error. The criticality index for activity 4-3 represents total number of real-

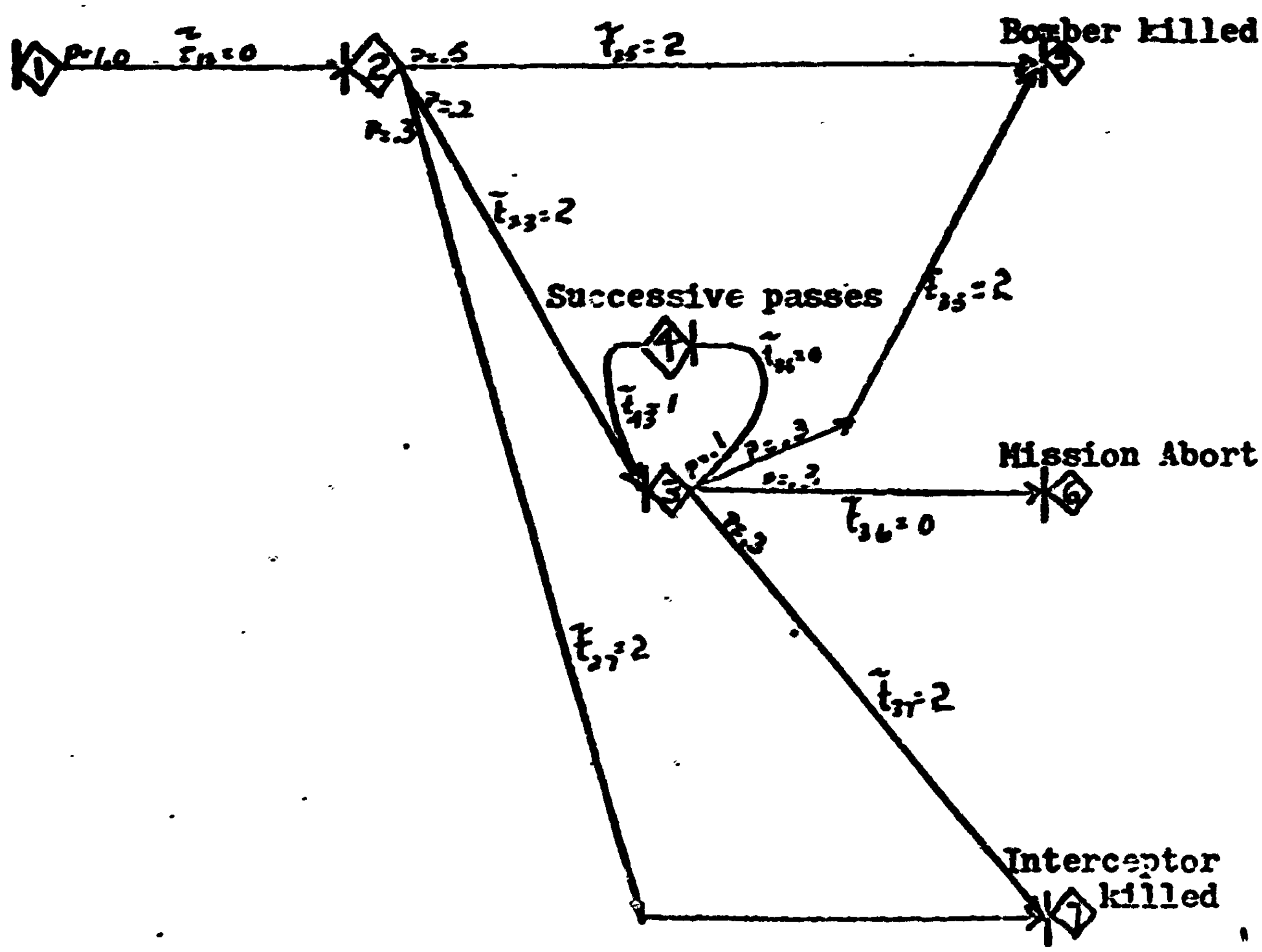


FIGURE 22

GERT NETWORK FOR BOMBER INTERCEPTOR PROBLEM

izations over the 1000 simulation trials, and is used as a simulation diagnostic and not a measure of criticalness.

TABLE III

SIMULATED CRITICALITY INDEX FOR
ACTIVITIES OF THE BOMBER INTERCEPTOR PROBLEM

Activity	I	J	CRITICALITY
	1	2	1.00
	2	3	0.21
	2	5	0.48
	2	7	0.31
	3	4	0.00
	3	5	0.08
	3	6	0.08
	3	7	0.05
	4	3	0.04

Node Analysis was conducted on nodes 5, 6, and 7, for probability of realization and distribution of realization time. Results of the network simulation as contrasted with those obtained through application of the analytic GERT computer program are summarized in Table IV. A test of proportions, based on the normal approximation and Z test statistic previously utilized, is applied to the probability of realization obtained for each node. The number of simulation trials was 1000.

TABLE IV

RESULTS OF NODE ANALYSIS ON BOMBER INTERCEPTOR PROBLEM FOR PROBABILITY AND EXPECTED TIME OF REALIZATION

	Probability of Realization		Expected Time		
Nodes	Simulated	Analytical	Simulated	Analytical	Z Score
5	.563	.567	2.332	2.248	-0.19
6	.080	.067	2.112	2.111	1.64
7	.357	.367	2.272	2.384	-0.65

There is no reason to reject the hypothesis of equality of simulated and analytical probabilities of realization at the 5 per cent confidence level. Although the Z score for node 6 is high, it falls within the acceptable confidence limits of $Z = \pm 1.96$.

The histograms for time of realization of nodes 5, 6, and 7 are given in Figure 23 . While a complete cell by cell comparison of expected versus observed histogram cell size, utilizing the X^2 statistic, would provide the strongest test of the accuracy of the simulation results, it is felt that comparison on one cell from each histogram would provide sufficient information. The test of proportions is applied to the time of realization $t < 3$ for each node and is shown in Table V.

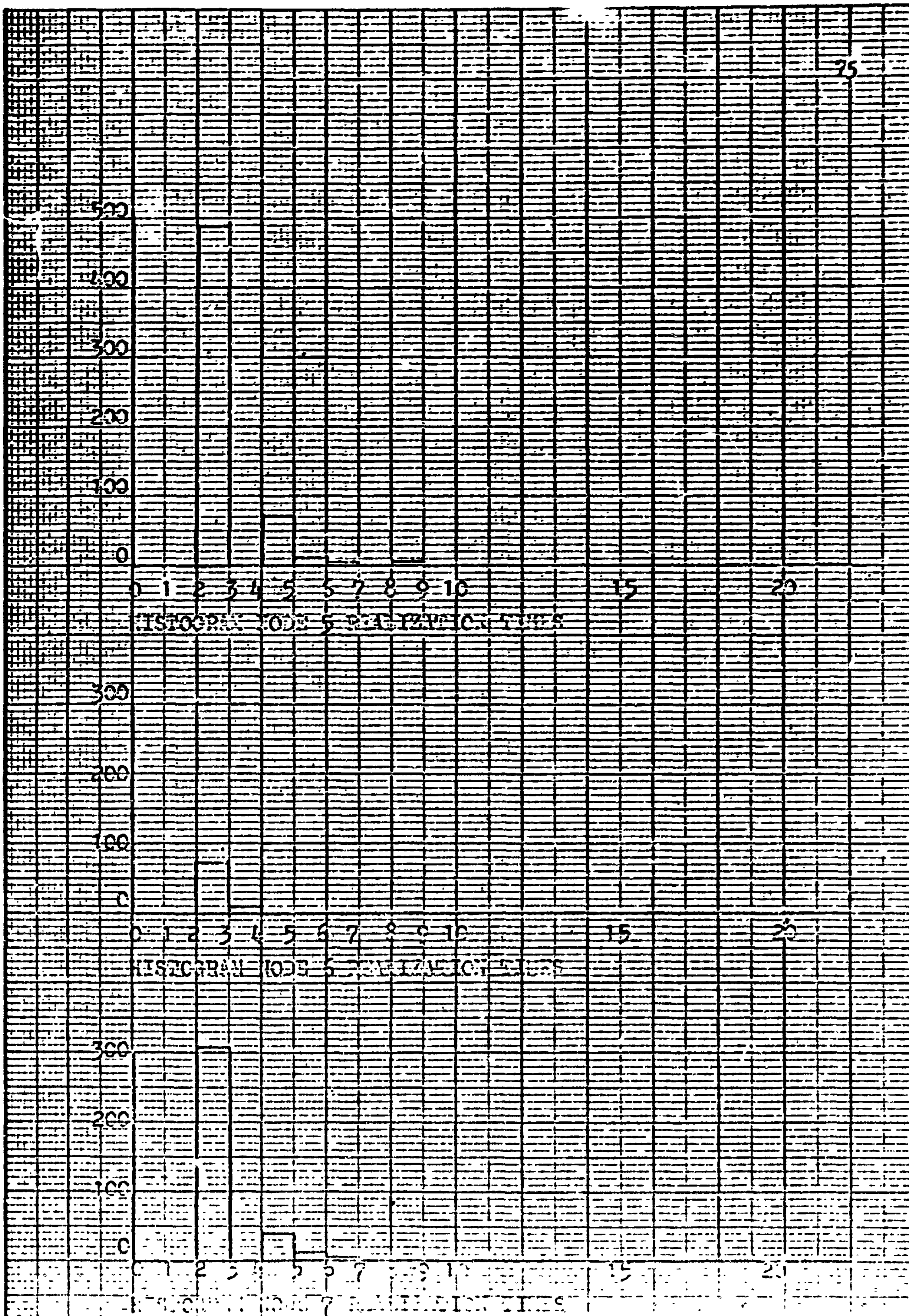


FIGURE 23

HISTOGRAMS FROM MODE ANALYSIS OF [illegible]

TABLE V

ANALYSIS OF TIME OF REALIZATION ($t<3$)
NODES 4, 5, AND 7, OF THE BOMBER INTERCEPTOR PROBLEM

Node	Simulated	Analytical	Z Score
5	.480	.50	-1.26
6	.071	.06	1.46
7	.312	.30	.83

The test of the sample time values for all three nodes indicates no significance at the 5 per cent confidence level. However, the high score for nodes 5 and 6 indicate more simulations of the network are necessary to establish more accurate results.

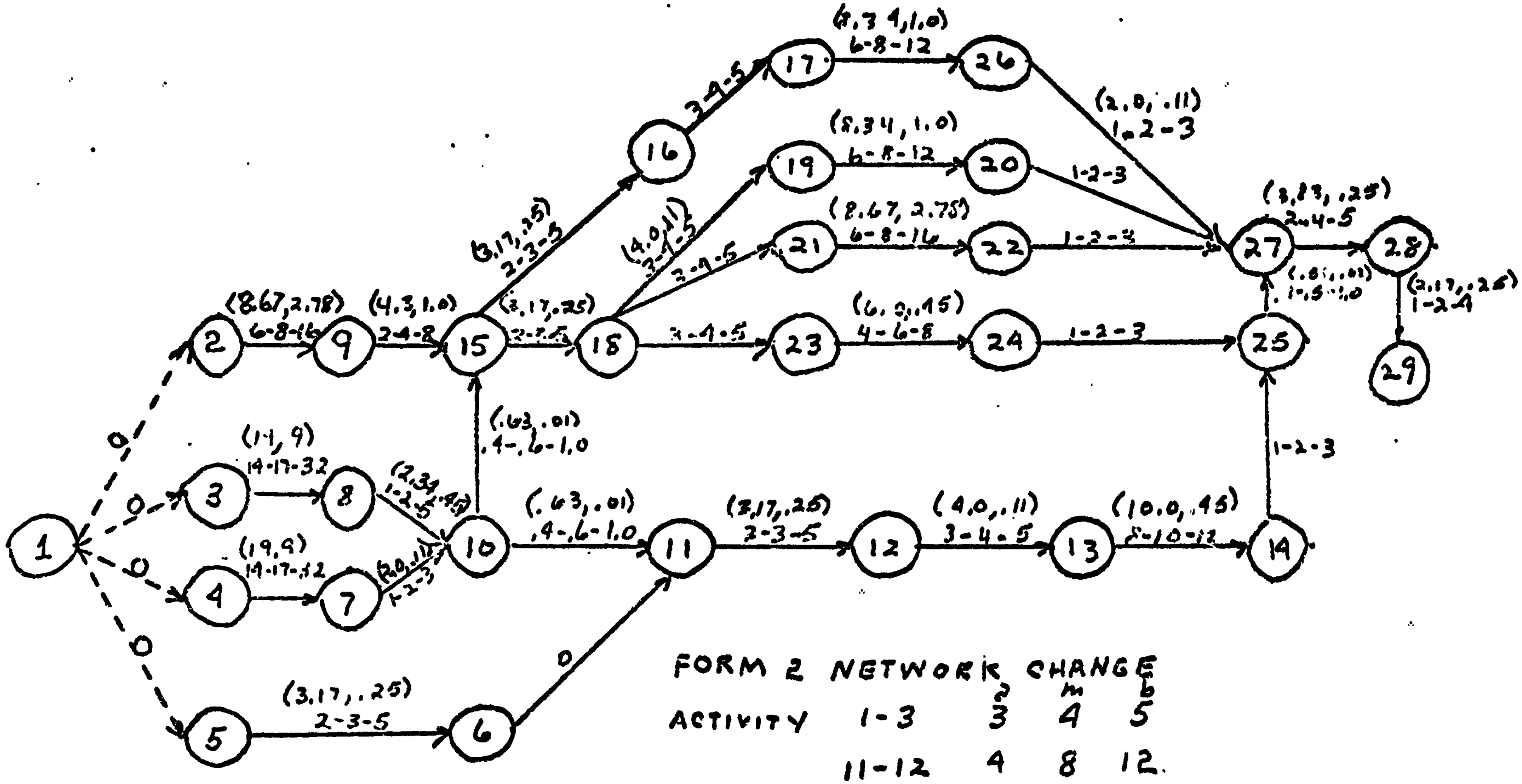
II. DETERMINISTIC NETWORKS

The network chosen to exhibit solution of deterministic GERT networks, consisting of all AND nodes, also serves a dual purpose of investigating PERT networks and related theory. The network is depicted in Figure 24 with activity duration times given as the PERT a, m, and b, time estimates. Also included above each time estimate in the network diagram is the mean and variance calculated from the PERT approximation formulas:

$$u_x = E(x) = (a + 4m + b)/6$$

$$VAR(x) = (b - a)^2/36 \quad .$$

This network was simulated using each of two probability distributions, Beta and Normal, and in two configurations or forms. Form 1 contains the network and activities as given by Moder and Phillips and



ACTIVITY	a	m	b
1-3	3	4	5
11-12	4	8	12
14-25	3	4	5

FIGURE 24. PERT DIAGRAM, FORM 1 AND FORM 2

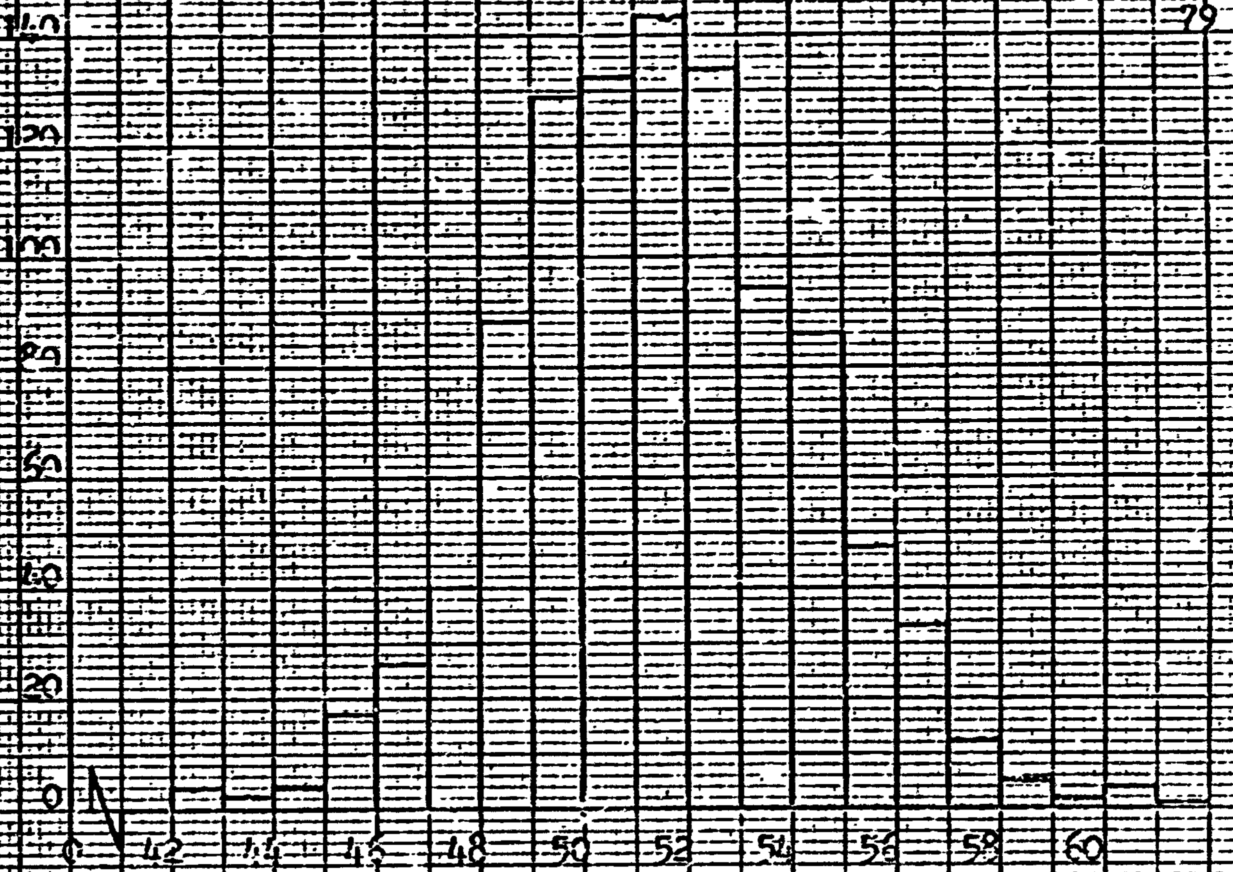
presented in Figure 24.[7] Form 2 is a modified form 1 network with the time estimates for 3 activities lengthened to establish one dominant critical path through the network. These activities are also noted on the network diagram. The major part of this discussion will pertain to the network referred to as form 1.

Although the approximation formula for activity mean and variance were derived from the Beta distribution, PERT solutions usually consider activity duration to be normally distributed, as well as project duration. Simulation of this network (form 1) was accomplished using both Normal and Beta distributions for 1000 trials.

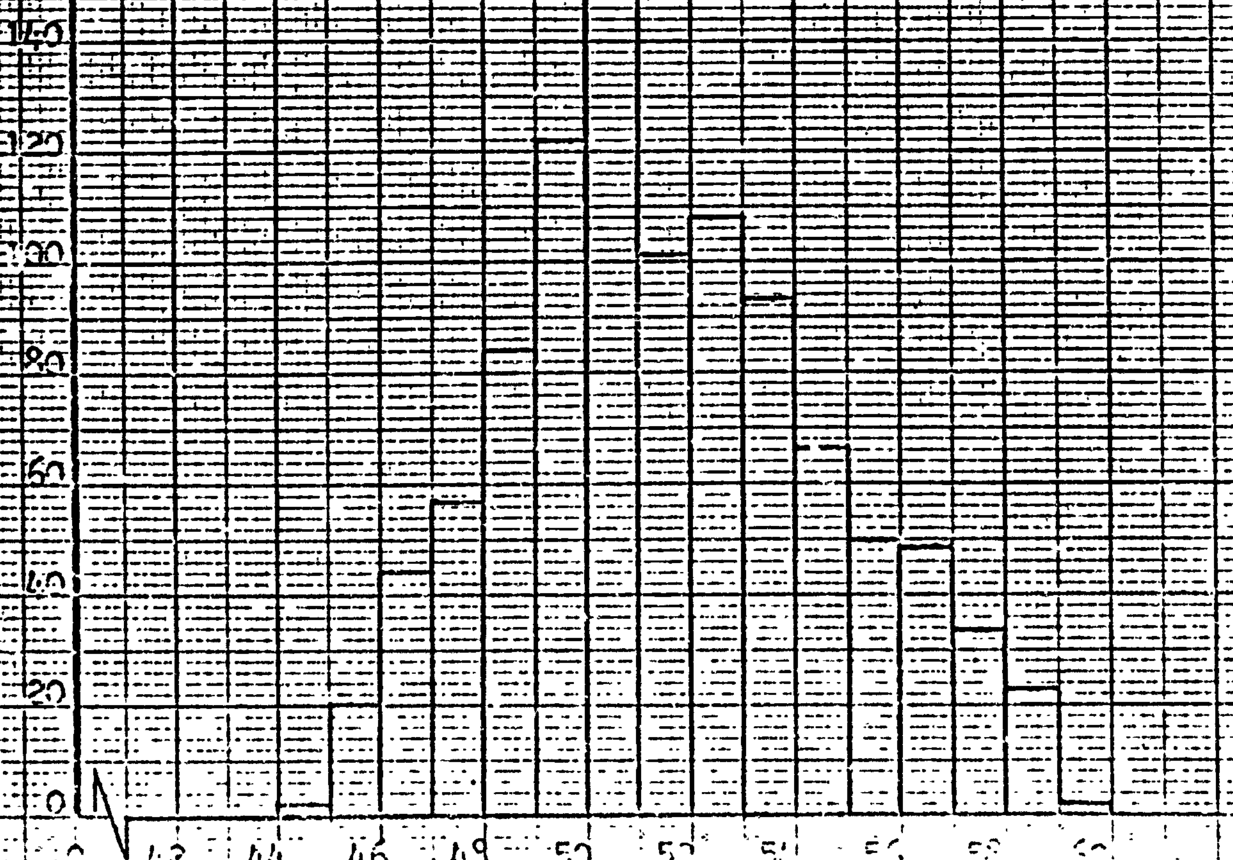
Histograms of the resultant distributions for the two simulation runs are given in Figures 25 and 26. The Chi Square Goodness of Fit Test was applied comparing the simulated resultant distribution with the tabulated Normal Density Function. The results were so inconclusive, that another simulation using the Beta distribution was accomplished for 9999 trial simulations. A histogram of the resultant distribution of network duration for the 9999 simulation trials is given in Figu: 7.

---

[7] Joseph J. Moder and Cecil R. Phillips, Project Management With CPM and PERT (New York: Reinhold Publishing Company), p. 212.

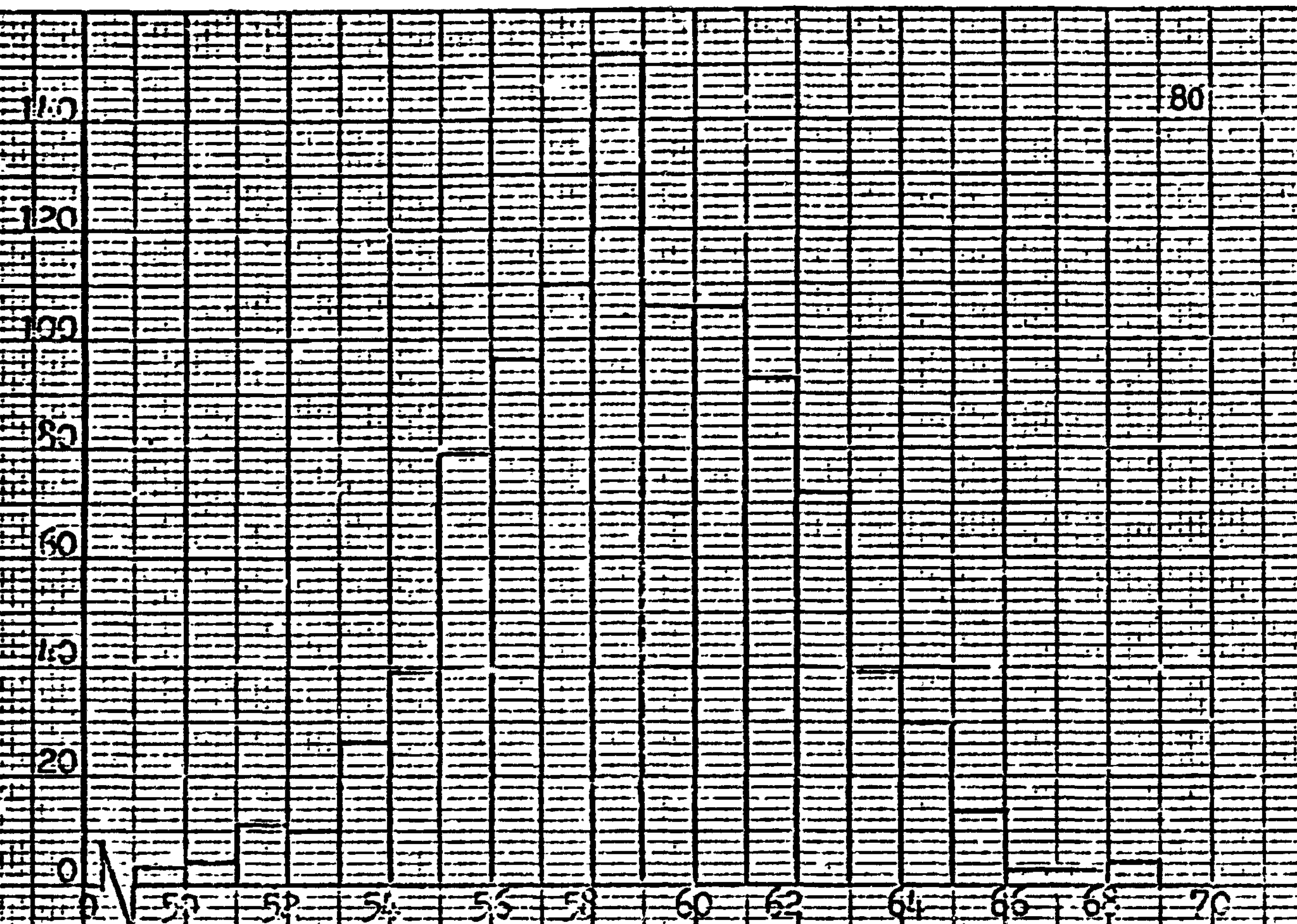


A. HISTOGRAM OF FORM 1 NETWORK, NORMALLY DISTRIBUTED ACTIVITIES

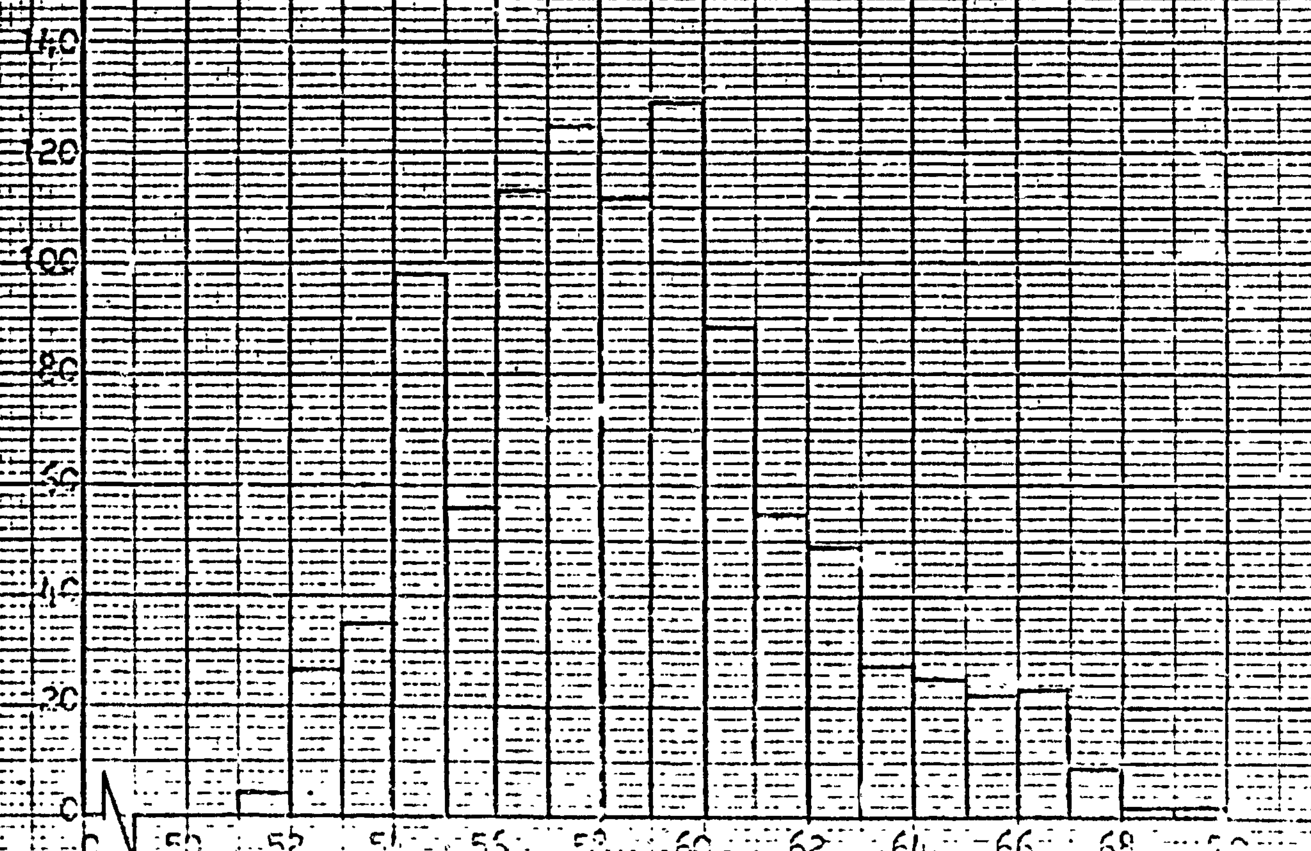


B. HISTOGRAM OF FORM 1 NETWORK, BETA DISTRIBUTED ACTIVITIES

FIGURE 25. NETWORK REALIZATION TIMES, [illegible] NETWORK
1000 SIMULATION TRIALS



C. HISTOGRAM OF FORM 2 NETWORK, NORMALLY DISTRIBUTED ACTIVITIES



D. HISTOGRAM OF FORM 2 [illegible], [illegible]A DIS[illegible] [illegible]

FIGURE 26. NETWORK REALIZATION TIMES, PERT NETWORK
1000 SIMULATION TRIALS

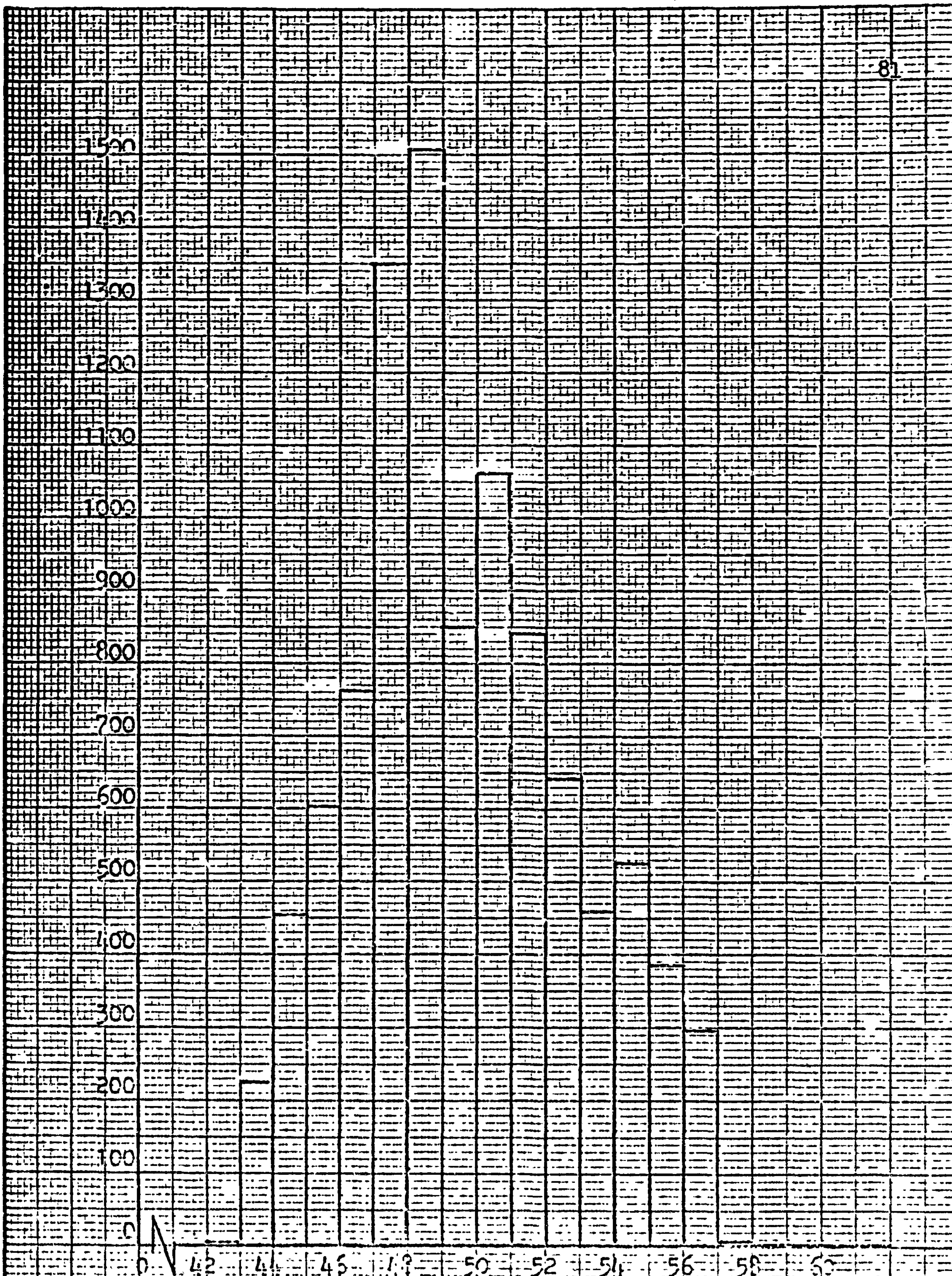


FIGURE 27. HISTOGRAM OF FORM 1 NETWORK REALIZATION TIMES, WITH BETA DISTRIBUTED ACTIVITIES, FOR 9999 SIMULATION TRIALS

TABLE VI

RESULTS OF THE SIMULATION
OF THE PERT NETWORK

Network		Form 1		Form 2	
Distribution	Beta	Beta	Normal	Beta	Normal
Number of Trials	1000	9999	1000	1000	1000
Mean	49.6036	49.6248	49.5002	58.6316	58.9318
Variance	9.8428	10.2629	7.9085	11.6842	9.8154
X^2 Value	92.3920	1435.6596	25.1858	114.8033	14.9355

While the expected values (form 1 network) are all essentially the same, the variances show a slight increase from Normal to Beta distributions for 1000 trials to the Beta 9999 trials. The X^2 values for both simulation runs of the form 1 network with the Beta distribution are highly significant indicating that normality of networks containing Beta distributed activities could not be assumed. However, the X^2 value for the simulation using normally distributed activity times indicates acceptance of the resultant distributions as normal, since

$$X^2 = 25.1858 < X^2_{.95,17} = 27.587$$

The form 2 network was simulated to determine if a network with one dominant critical path would improve upon the results of the preceding paragraph. Three activities were lengthened. The results of the 1000 simulation trials utilizing the Beta and Normal distributions are also given in Table VI. While the X^2 value for the Beta distributed form 2 network increased to a value more highly significant than that of the form 1 network, for 1000 trials, the Normally distributed network

improved, as indicated by a decreased in the value of X^2.

The criticality index for all simulation runs of the PERT network (forms 1 and 2) is given in Table VII, indicating the relative importance of the activities within the network contributing to network realization. There are two major critical paths with several activities in common contributing to network realization for the form 1 network.

Comparison, of the information given by the criticality index and resultant distributions of network realizations, reveals that both forms of the network containing Beta distributed activity times produce multi-modal network distributions. The resultant distribution of the form 1 network, where there is more than one critical path with activities in common, more closely approaches normality than the resultant distribution of the form 2 network, where only one critical path is present.

Another factor of importance concerns use of the PERT assumptions in determining network completion time defined as the sum of expected activity durations on the critical path of the network. The critical path is the path of maximum sum of expected activity durations in the network from origin to terminal nodes. Based on this assumption, the expected time of the form 1 network is 47.65 and variance is 10.9. Comparison, with the simulated results, indicates the PERT estimate is optimistic, consistent with the statement made by Fulkerson that PERT produces optimistic estimates, such that the estimate is greater than the expected value.[8]

However, the PERT assumption of normally distributed network real-

---

[8] D. R. Fulkerson, *Expected Critical Pat[illegible] Networks*, The Rand Corporation, RM-3075-PR, (Santa Monica, March, 19[illegible]) p. 1.

TABLE VII

CRITICALITY INDEX FOR ALL SIMULATION RUNS OF THE PERT NETWORK ANALYSIS

Activity		Criticality				
		Form 1 Network			Form 2 Network	
I	J	Beta	Beta*	Normal	Beta	Normal
1	2	0.00	0.00	0.00	0.00	0.00
1	3	0.52	0.51	0.53	0.81	0.84
1	4	0.48	0.49	0.46	0.19	0.16
1	5	0.00	0.00	0.00	0.00	0.00
2	9	0.00	0.00	0.00	0.00	0.00
3	8	0.52	0.51	0.53	0.81	0.84
4	7	0.48	0.49	0.46	0.19	0.16
5	6	0.00	0.00	0.00	0.00	0.00
6	11	0.00	0.00	0.00	0.00	0.00
7	10	0.48	0.49	0.46	0.19	0.16
8	10	0.52	0.51	0.53	0.81	0.84
9	15	0.00	0.00	0.00	0.00	0.00
10	11	0.65	0.63	0.72	1.00	1.00
10	15	0.35	0.37	0.28	0.00	0.00
11	12	0.65	0.63	0.72	1.00	1.00
12	13	0.65	0.63	0.72	1.00	1.00
13	14	0.65	0.63	0.72	1.00	1.00
14	25	0.65	0.63	0.72	1.00	1.00
15	18	0.26	0.27	0.21	0.00	0.00
15	16	0.08	0.10	0.06	0.00	0.00
16	17	0.08	0.10	0.06	0.00	0.00
17	26	0.08	0.10	0.06	0.00	0.00
18	29	0.05	0.05	0.06	0.00	0.00
18	21	0.20	0.22	0.16	0.00	0.00
18	23	0.00	0.01	0.00	0.00	0.00
19	20	0.05	0.05	0.06	0.00	0.00
20	27	0.05	0.05	0.06	0.00	0.00
21	22	0.20	0.22	0.16	0.00	0.00
22	27	0.20	0.22	0.16	0.00	0.00
23	24	0.00	0.01	0.00	0.00	0.00
24	25	0.00	0.01	0.00	0.00	0.00
25	27	0.66	0.64	0.72	1.00	1.00
26	27	0.08	0.10	0.06	0.00	0.00
27	28	1.00	1.00	1.00	1.00	1.00
28	29	1.00	1.00	1.00	1.00	1.00

*This column represents data for the 9999 Simulation trials of the Beta distributed network. All other data is for 1000 trials.

ization times utilizing the approximation formula for mean and variance does hold for the network analyzed.

This chapter has presented applications of the GERT Simulation model through simulation of various GERT networks. The PERT assumptions were investigated through analysis of GERT networks composed of all AND nodes. The networks utilized to test probabilistic GERT applications required approximately one minute each for 1000 simulation trials on the CDC 3400 computer. The PERT networks, utilizing the Beta probability distribution required approximately 4 minutes of computer time for 1000 simulation trials.

CHAPTER V

ANALYSIS OF AND NODES

The results of a GERT network analysis are the probability of node realization, and the distribution of the equivalent time parameter for that node given it is realized. Extensive research has been conducted on networks containing EXCLUSIVE-OR logic nodes, while conceptual and computational problems exist in analysis of networks containing the AND logic node.

Realization of the AND logic node occurs when all the activities inwardly-directed to the AND node have been completed or realized. Consideration of the probability of node realization is necessary since GERT networks may be composed of probabilistic and deterministic type logic nodes. Statements about the associated time parameters and distributions are then conditioned on the probability of realizing the node. Should the network consist of all AND nodes, as in PERT, the probability of node realization is one. The time at which a node is realized is the maximum of the durations of the inwardly directed paths to that node.

The purpose of this chapter is to investigate general applicability of the merge bias correction technique proposed by Clark[1] in analysis of GERT networks containing AND nodes. A technique is required to compensate for erroneous results stemming from analysis based on expected values, as in PERT, and to reduce computational difficulties associated

---

[1]Charles E. Clark, "The Greatest of a Finite Set of Random Variables," Op. Res., Vol. 9, No. 2, 1961, pp. 145-162.

with the maximization operation. Pre. ntation of this chapter is by development of the theory and through application on two sample networks.

The merge bias correction procedure is concerned with providing the moments of the maximum of random variables where the random variables have a joint probability distribution. Networks composed of activities with independent distributed variables for time duration provide an application for this procedure.

The basic assumption of this procedure is that the independent activity random variables w, x, y, and z are normally distributed with expected value u_w and variance σ^2_w and $\sigma^2_{\cdot}$. Linear correlation may exist between any two paths, such as x and y, and is denoted by r(x,y) or $\rho_{x,y}$. The notation $\phi(x) = 1/\sqrt{2\pi}\,\exp(-x^2/2)$, and $\Phi(x) = \int_{-\infty}^{x}\phi(t)dt$ are used for the normal distribution. Also, max(x,y) is denoted by x:y

Let v_i be the i^{th} moment about zero of the random variable max(x,y). The following equations are derived by Clark and are utilized in solving for the maximum of random variables.

$$a^2 = \sigma_x^2 + \sigma_y^2 - 2\sigma_x\sigma_y\rho_{x,y}, \quad \text{where } \sigma_x \neq \sigma_y,\ \rho_{x,y} \neq 1.$$

$$\beta = (u_x - u_y)/a$$

$$v_1 = u_1\,\Phi(\beta) + u_2\,\Phi(-\beta) + a\,\phi(\beta)$$

$$v_2 = (u_1^2 + \sigma_1^2)\,\Phi(\beta) + (u_2^2 + \sigma_2^2{}_{-}\ \Phi(-\beta) + (u_1 + u_2)\,a\,\phi(\beta)$$

The linear correlation coefficient is given approximately as

$$r(z,\max(x,y)) = \rho_{z,x:y}$$

$$= (\sigma_x\rho_{x,z}\,\Phi(\beta) + \sigma_y\rho_{y,z}\,\Phi(-\beta))\,/\,\sigma_{x:y}.$$

Another special form of the correlation coefficient is also useful.

$$r(X + x, Y + y) = \rho_{X+x,Y+y}$$

$$= \sigma_x \sigma_y \rho_{x,y} / \sigma_{X+x} \sigma_{Y+y}$$

The equations given above may be utilized in obtaining the maximum of any number of random variables by consideration of pairs of variables through successive application of the following procedure.

$$\text{Max}(w, x, y, z) = \text{Max}\left\{ w, \max(x, \max(y,z)) \right\}$$

At this point, perhaps a simple example would clarify the discussion. Consider the network shown in Figure 28 . The probability of realizing node 2 is (.3) (.4) (.2) (.8) (.5) = 0.0096 due to the combination of AND and EXCLUSIVE-OR nodes as constructed.[2]

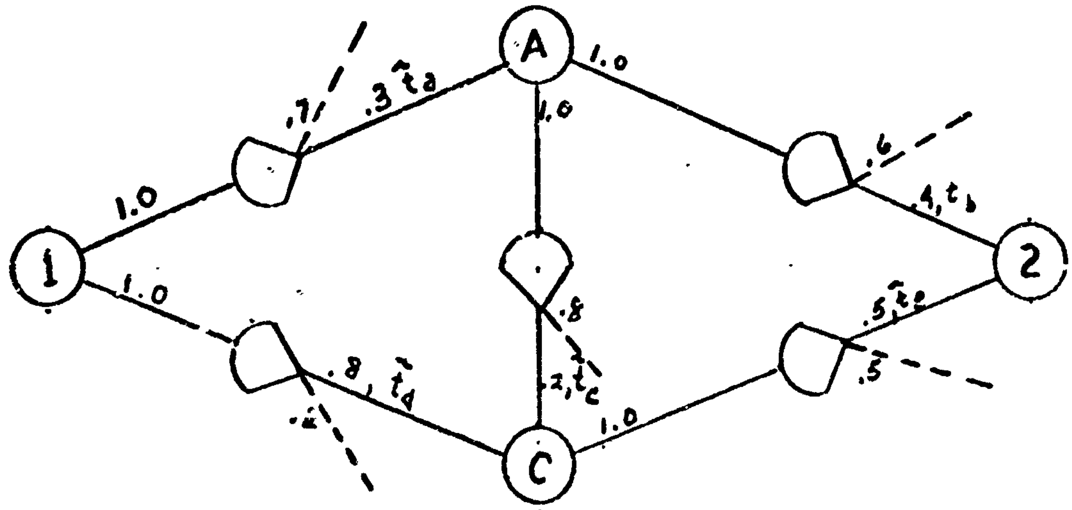


FIGURE 28

SAMPLE NETWORK UTILIZING AND LOGIC NODES

The time of realization of node 2, T_2, given it is realized is the

---

[2] A. Alan B. Pritsker, GERT: Graphical Evaluation and Review Technique, the RAND Corporation, [illegible] (Santa Monica, April, 1966), Appendix [illegible].

maximum of the 3 paths leading from node 1 to node 2 as follows. The notation $\tilde{t}_i$ represents the random variable, Time of duration for activity i.

$$\tilde{T}_2 = \mathrm{Max}\,(\tilde{t}_a + \tilde{t}_b,\ \tilde{t}_a + \tilde{t}_c + \tilde{t}_e,\ \tilde{t}_d + \tilde{t}_e)$$

$$\tilde{T}_2 = \mathrm{Max}\,(\tilde{t}_a + \tilde{t}_b,\ \tilde{t}_e + \mathrm{Max}\,(\tilde{t}_a + \tilde{t}_c,\ \tilde{t}_d))$$

Traversing the network for $\tilde{T}_2$, assuming node 1 has a mean of zero and variance of one:

Node A: $\tilde{T}_A = \tilde{t}_a$, with parameters Mean(A), and Var(A).

Node C: $\tilde{T}_C = \mathrm{Max}\,(\tilde{T}_A + \tilde{t}_c,\ \tilde{t}_d)$

$\rho_{a+c,d} = 0$, since a+c and d are independent parallel paths.

$$u_{a+c} = u_a + u_c$$

$$u_d = u_d$$

$$a^2 = \sigma_d^2 + \sigma_{a+c}^2$$

$$\beta = (u_d - u_{a+c}) \,/\, (\sigma_d^2 + \sigma_{a+c}^2)$$

$$v_1 = u_d\,\Phi(\beta) + u_{a+c}\,\Phi(-\beta) + a\,\phi(\beta)$$

$$v_2 = (u_d^2 + \sigma_d^2)\,\Phi(\beta) + (u_{a+c}^2 + \sigma_{a+c}^2)\,\Phi(\beta) + (u_d + u_{a+c})\,a\,\phi(\beta).$$

$$\mathrm{Mean}\,(C) = v_1$$

$$\mathrm{Var}\,(C) = v_2 - v_1^2$$

Node 2: $\tilde{T}_2 = \text{Max}\,(\tilde{T}_A + \tilde{t}_b\,,\ \tilde{T}_C + \tilde{t}_e)$

$$u_{A+b} = \text{Mean}(A) + u_b$$

$$= u_a + u_b$$

$$u_{C+e} = \text{Mean}(C) + u_e$$

$$\sigma^2_{A+b} = \text{Var}(A) + \sigma^2_b$$

$$\sigma^2_{C+e} = \text{Var}(C) + \sigma^2_e$$

$$\rho_{A+b,C+e} = \sigma_A\,\sigma_C\,\rho_{A,C} / \sigma_{A+b}\,\sigma_{C+c}$$

$$\rho_{A,C} = \rho_{a,d:a+c}$$

$$= \frac{\sigma_d\,\rho_{a,d}\,\Phi(\beta) + \sigma_{a+c}\,\rho_{a,a+c}\,\Phi(-\beta)}{\sigma_{d:a+c}}$$

The correlation coefficient, $\rho_{a,d} = 0$, since a and d are independent activities in the network.

$$\rho_{a,a+c} = \sigma_a\,\sigma_a\,\rho_{a,a} / \sigma_a\,\sigma_{a+c}$$

$$= \sigma_a / \sigma_{a+c}$$

$$\rho_{A,C} = \sigma_{A+c}\,\sigma_a\,\Phi(-\beta) / \sigma_{d:a+c}$$

$$\rho_{A+b,D+e} = \sigma_A\,\sigma_C\,\sigma_{A+c}\,\sigma_a\,\Phi(-\beta) / \sigma_{A+b}\,\sigma_{C+e}$$

Thus calculation of a new a, β, v_1, and v_2, for node 2, is possible when $\rho_{A+b,D+e}$ is known. Mean(2) and Var(2) follow from these calculations.

The equivalent function $W_e(s)$ for the network would be,

$$w_e(s) = p_{12} \exp(\text{Mean}(2)s + \tfrac{1}{2}\text{Var}(2)s^2)$$

where p_{12} is the probability of realizing node 2, calculated at the beginning of this chapter.

Another procedure would be to add pseudo activities in the network at compensating points to visually indicate bias corrections affecting network calculations and allow conventional (PERT) network solution using expected times.[3] The pseudo activities would in a sense be non-zero dummy activities emanating from the affected node with mean and variance being the difference between merge bias and conventional calculated mean and variance.

A second, more complex, network with numerical values is offered to further illustrate calculations involved and simplifying shortcuts which may be used to decrease the computational workload. Since the equations utilized are the same as the first sample network, only a brief coverage will be given the actual calculations. The GERT Simulation Program was utilized as a check against hand calculation and offers further insight into solution of the network. Comparison of results follows hand calculations.

The network is as shown in Figure 29; where the activities are a, b, c, $\cdots$,k; node numbers are 1, 2, 3, $\cdots$,6; and mean and variance are denoted as (u, σ^2) respectively. The probability of node realization is one, so all activities must be realized for network completion.

---

[3]Joseph J. Moder and Cecil R. Phillips, Project Management with CPM and PERT (New York: Reinhold Publishing Co., 1964), p. 237.

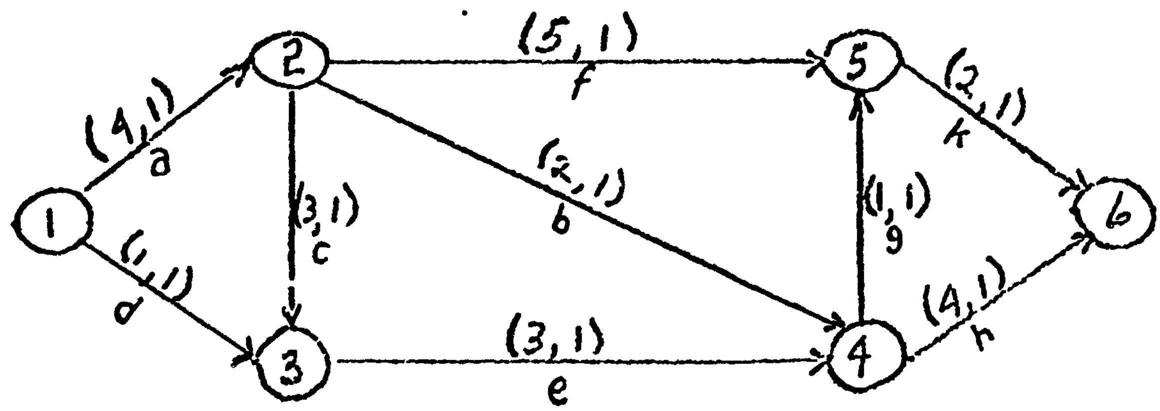


FIGURE 29
PERT NETWORK UTILIZED IN AND NODE ANALYSIS

Node 1: Mean(1) = 0 , Var(1) = 0

Node 2: Mean(2) = 4 , Var(2) = 1, since activity a is an independent activity.

Node 3: $\tilde{T}_3 = \text{Max}\ (\tilde{t}_a + \tilde{t}_c, \tilde{t}_d)$

$u_{a+c} = 7$, $u_d = 1$, $\sigma_d^2 = 1$

$\rho_{a+c,d} = 0$, since a + c and d are independent parallel paths with no common element.

$a^2 = 2 + 1 = 3$, $\beta = -3.46$

$v_1 = 6.997$, $v_2 = 50.9992$

Mean(3) = 7

Var(3) = 2

At this point a simplifying aspect is noted, in that Mean(3) and Var(3) are at most 1/10 per cent off the conventional calculations using expected values. A rule given by Moder and Phillips applies in this case:[4]

> If the difference between the expected complete times of the two merging activities being considered is greater than twice the larger of their respective standard deviations, then the bias correction will be small, less than a few per cent, and can be ignored.

Node 4: $\tilde{T}_4 = \text{Max}\ (\tilde{T}_3 + \tilde{t}_e, \tilde{T}_2 + \tilde{t}_b)$

$u_{2+b} = 6$, $u_{3+e} = 10$, $\sigma^2_{2+b} = 2$, $\sigma^2_{3+e} = 3$

$\rho_{2+b,3+e} = .41$

$a^2 = 3$, $\beta = 2.31$

---

[4] *Ibid.*, p. 239.

$v_1 = 9.99$, $v_2 = 102.92$

Mean (4) = 9.99, Var(4) = 2.92

Here the rule also holds, but to a lesser effect, with the bias correction at .01 and .08 for the mean and variance respectively.

Node 5: $\tilde{T}_5 = \text{Max}\ (\tilde{T}_4 + \tilde{t}_g, \tilde{T}_2 + \tilde{t}_f)$

$u_{2+f} = 9,\ u_{4+g} = 11,\ \sigma^2_{2+f} = 2,\ \sigma^2_{4+g} = 3.92$

$\rho_{2+f,4+g} = .439$

$a^2 = 2.93,\ \beta = 1.17$

$v_1 = 11.1$, $v_2 = 126.8$

Mean(5) = 11.1, Va (5) = 4.78

Node 6: $\tilde{T}_6 = \text{Max}\ (\tilde{T}_4 + \tilde{t}_h, \tilde{T}_5 + \tilde{t}_k)$

$u_{4+h} = 13.99,\ u_{5+k} = 13.1$, $\sigma_{4+h} = 3.93$

$\sigma_{5+k} = 5.78$

$\rho_{4+h,5+k} = .286$

$a^2 = 7.0$, $\beta = .336$

$v_1 = 14.68,\ v_2 = 218.79$

Mean(6) = 14.68, Var(6) = 3.29

The sample network was also simulated using the GERT Simulation Program. All activities were normally distributed with mean and variance as indicated in Figure 29 on page 92 . The number of simulation trials

was 1000. A favorable comparison of the hand calculations and simulation results is given as follows for nodes 4 and 6.

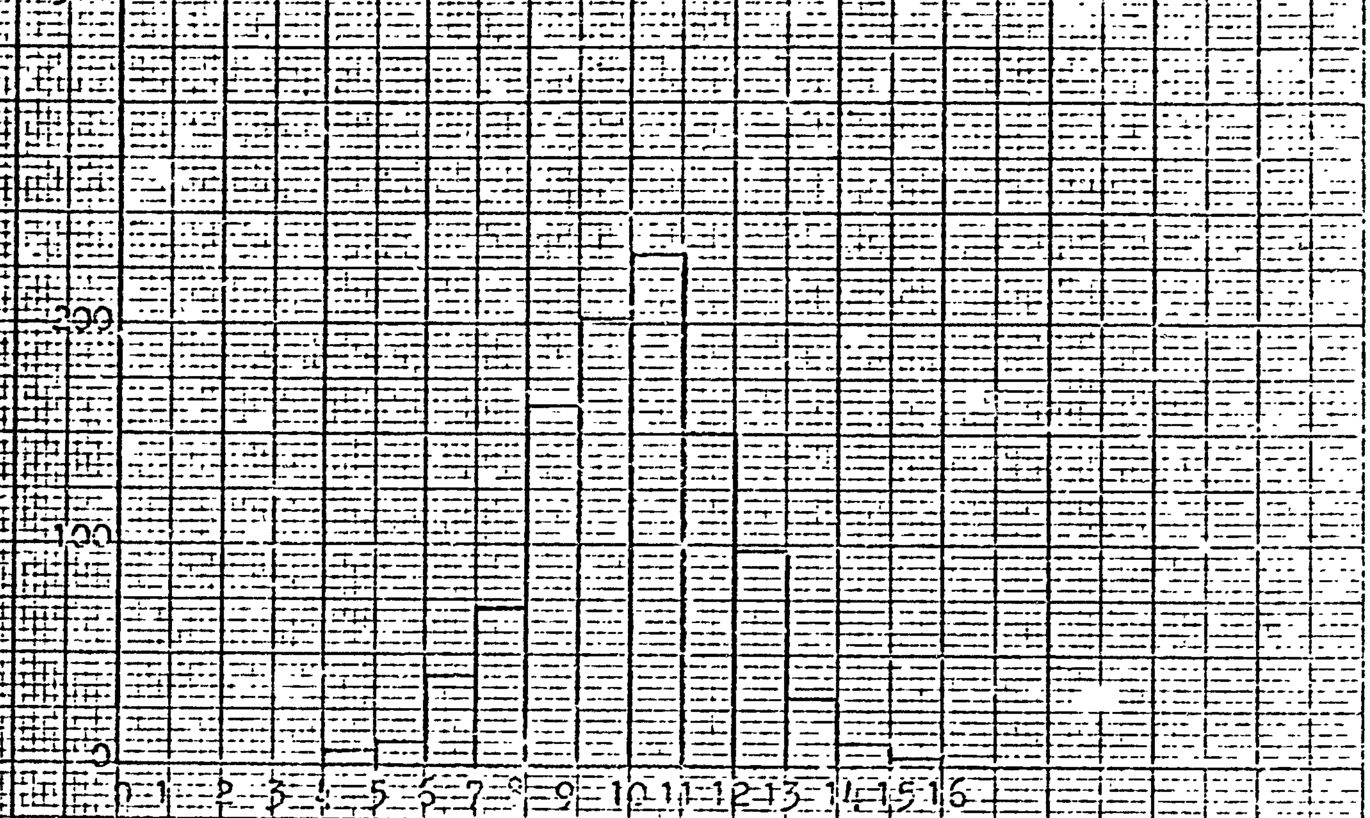
Node		4	6
Simulation	Mean	10.0487	14.3505
	Variance	3.0128	3.6519
Hand calculation	Mean	9.99	14.68
	Variance	2.92	3.29

The criticality index indicates one path is predominant, approximately 64 per cent of the trials, through the network from nodes 1, 2, 3, 4, to node 6. A minor critical path from nodes 1, 2, 3, 4, 5, to node 6, occurs approximately 31 per cent of the time. A histogram of node realization times for nodes 4 and 6 is given in Figure 30.

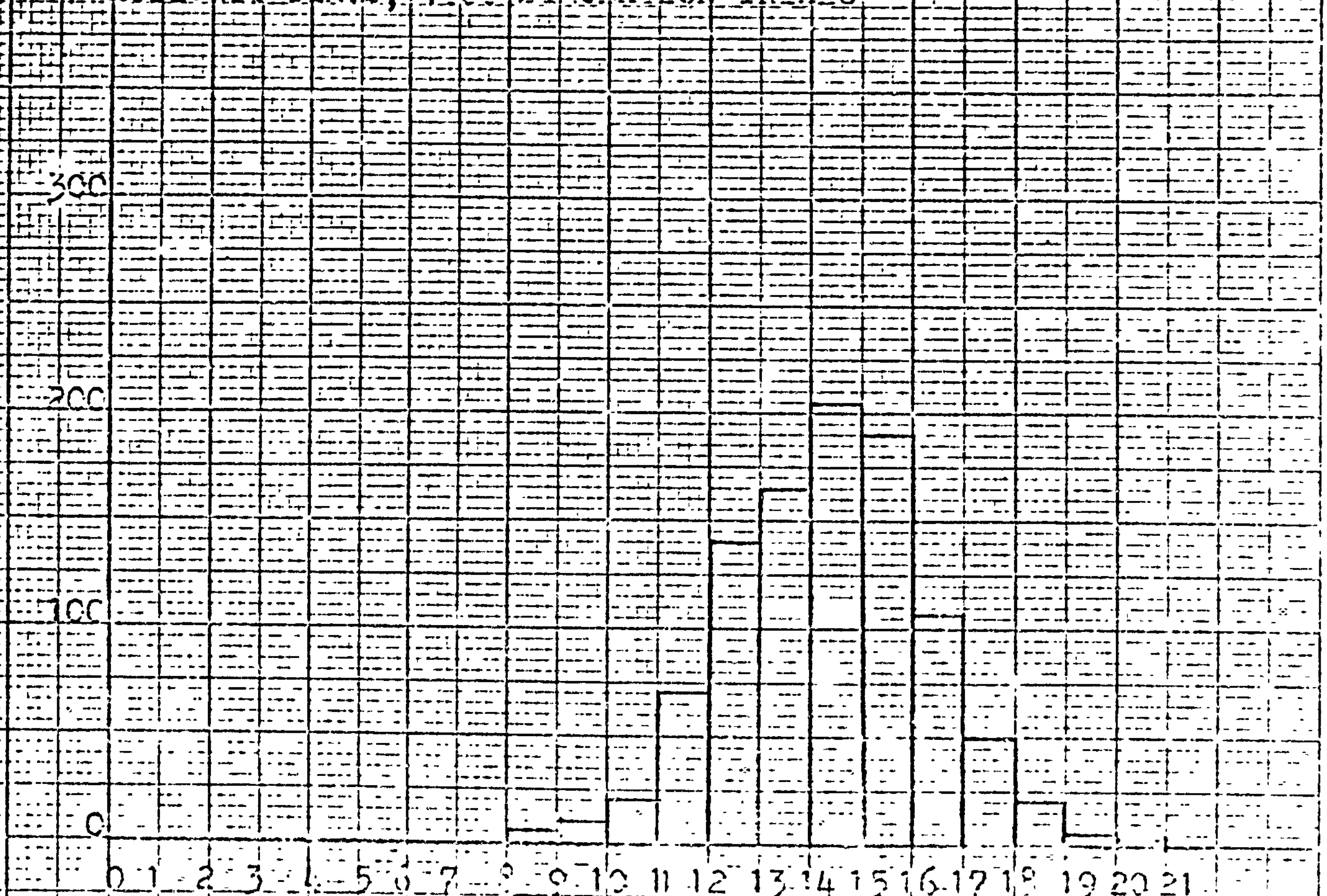
The assumption of normality for non-normal activities introduces error into the calculation and is discussed at length in the referenced article by Clark.[5] Approximation of the first two moments v_1 and v_2 of Max(x,y), a non-normal distribution, with the first two moments of a normal distribution does not produce too great an error when the expected values and variances are the same, for both distributions.

An example is given where the greatest of 500 standard normal variables, which is not normally distributed, is approximated by a normal variable with negligible error. Approximations of exponentially and uniformally distributed random variables by normally distributed variables are also exhibited resulting in a larger, but tolerable error.

[5] Clark, op. cit., p. 151.



HISTOGRAM OF NODE 4 REALIZATION TIMES, NORMALLY DISTRIBUTED ACTIVITY TIMES, 1000 SIMULATION TRIALS



HISTOGRAM OF NODE 6 REALIZATION TIMES, NORMALLY DISTRIBUTED ACTIVITY TIMES, 1000 SIMULATION TRIALS

FIGURE 30

HISTOGRAMS OF PERT NETWORK NODE REALIZATION TIMES, NODES 4 AND 6

The procedure of merge bias correction does provide a means for overcoming errors inherent in conventional network calculations based on expected values as in PERT. The calculations are tedious for hand calculation, but adaptable to computer solution.

The procedure is applicable to GERT network analysis, where the sub-networks containing AND nodes may be reduced to equivalent functions. Should the complexity of the network between AND nodes prevent network reduction, addition of the pseudo activities to the network would facilitate solution by the maximization operation using expected values. However, general applicability of merge bias correction cannot be claimed for GERT networks, as this procedure is mainly problem oriented.

CHAPTER VI

SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

I. SUMMARY

The purpose of this research was to: (1) develop a general purpose GERT Simulator allowing analysis of GERT networks through the technique of Simulation, (2) investigate resultant distributions of GERT networks containing all AND logic nodes to determine validity of the PERT normality assumptions, and (3) investigate analysis of AND logic nodes through analytical and simulation methods for distribution of the equivalent time parameter.

The goal to develop a generalized technique for analysis of complex systems portrayed by stochastic networks has not been fully realized. The general purpose GERT Simulation Program was needed to assist the research endeavor and reduce the computational complexity in solution of large networks.

The Simulation Model, was designed to facilitate solution of GERT networks, allowing as much freedom as possible in depicting networks for analysis. The program is user-oriented, as input data is held to a minimum, requiring only one input card for each activity in the network. Five probability density functions are available for use in describing the distribution of activity durations. Program output includes a criticality index on each activity as the relative frequency an activity appears on the critical path. The mean and variance of network completion is also provided as output, along with analysis of specified nodes, consisting of mean, variance, probability of realization and histograms of node realization times. A Chi-Square Goodness of Fit Test may be applied

to the distribution of node realization times and printed as output. Various GERT networks were simulated to provide examples of the use of the Simulation Model and verify results.

An investigation into the PERT normality assumptions was conducted through use of the Simulation Model. The Beta and Normal probability distributions were utilized in simulating a sample network. Resultant distributions of network realization times were analyzed to determine if normality could be assumed.

Analysis of AND logic nodes was conducted using simulation and analytical techniques. The merge bias correction procedure was applied to sample networks to determine if the procedure was applicable to solution of GERT networks containing AND nodes.

II. CONCLUSION

The GERT Simulation Model did satisfy all requirements set forth in Chapter I. Simulation of sample GERT networks provided logical and conclusive results as established by appropriate statistical tests.

The normality assumptions utilizing approximation formulas for mean and variance of activity duration in solution of PERT networks was verified by analysis of resultant network distributions. Strict application of the Beta probability distribution for activity duration times did not produce normally distributed network realization times for either of the network configurations considered.

The merge bias correction procedure utilized to produce better approximations of network realization times than conventional methods was determined to be computationally difficult for large networks,

although providing accurate estimates of distribution parameters. However, the normality assumption of activity distribution could introduce significant errors in estimation. This procedure is therefore problem-oriented and not applicable to general solution of GERT networks.

III. RECOMMENDATIONS

The use of the Simulation Model will facilitate further research into GERT networks. However, further applications, other than those presented in this paper, are necessary to assure satisfactory results in the general case, and are therefore recommended.

The application of the merge bias technique to general GERT network solution for networks containing AND logic nodes is computationally difficult. Further investigation into this procedure and the underlying normality assumptions is recommended. A computer program to solve PERT networks with this procedure would provide resultant statistics for comparison with simulated and analytical results.

BIBLIOGRAPHY

Bowker, Albert H., and Gerald J. Lieberman. Engineering Statistics. Englewood Cliffs: Prentice Hall, Inc., 1964.

Charnes, A., W. W. Cooper, and G. L. Thompson. "Critical Path Analysis via Chance Constrained and Stochastic Programming," Op. Res., Vol. 12, No. 3, 1964, pp. 460-470.

Clark, C. E. "The Greatest of a Finite Set of Random Variables," Op. Res., Vol. 9, No. 2, 1961, pp. 145-162.

______"The PERT Model for the Distribution of an Activity Time," Op. Res., Vol. 10, No. 9 1962, p. 405.

Deutsch, Don and Phillip Wolfe, "A Revision to Gasp, A Simulation Programming Language," unpublished term paper, IE 477g, Arizona State University, May, 1965.

Fulkerson, D. R. Expected Critical Path Lengths in PERT Networks, The Rand Corporation, RM-3075-PR. Santa Monica: March, 1962, p.l.

Harter, H. Leon. New Tables of the Incomplete Gamma Function Ratio and of Percentage Points of the Chi Square and Beta Distribution. Aerospace Research Laboratories, Office of Aerospace Research, United States Air Force. Washington: Government Printing Office, 1964, Table 3.

Kahn, Herman. Applications of Monte Carlo. The Rand Corporation, AECU-3259. Santa Monica: April, 1954, p. 10.

Kunz, Kaiser S., Numerical Analysis. New York: McGraw-Hill Book Company, Inc., 1957, p. 18.

MacCrimmon, K. R., and C. A. Ryavec, An Analytical Study of the PERT Assumptions. The Rand Corporation, RM- . Santa Monica: December, 1962.

McDonald, William W. "Network Analysis by Simulation." Industrial Engineering Research Bulletin, No. 2 Arizona State University, February, 1966.

McMillan, Claude, and Richard F. Gonzalez, Systems Analysis. Homewood, Illinois: Richard D. Irwin Inc., p. 157.

Moder, Joseph J, and Cecil R. Phillips, Project Management with CPM and PERT. New York: Reinhold Publishing Co., 1964, p. 237.

Pritsker, A. Alan B., GERT: Graphical Evaluation and Review Technique. The Rand Corporation, RM-4973-NASA. Santa Monica: April, 1966.

Special Projects Office, Bureau of Naval Weapons, Department of the Navy. PERT Summary Report, Phase 1. Washington: Government Printing Office, July, 1958.

Spiegel, Murray R. Statistics. New York: Schaum Publishing Co., 1961. p. 201.

Van Slyke, R. M. "Monte Carlo Methods and the PERT Problem," Op. Res., Vol. 11, No. 5, 1963, pp. 839-860.

BIOGRAPHICAL SKETCH

William Walton McDonald was born in Pueblo, Colorado, on May 22, 1937. He received his elementary and secondary education in the Bayless School District, St. Louis, Missouri. In 1955, he entered Washington University, St. Louis, Missouri, graduating in 1959 with a Bachelor of Science in Electrical Engineering. He entered the United States Air Force in 1959, receiving a commission of Second Lieutenant. He has served at United States Air Force Installations in Biloxi, Mississippi; Wichita, Kansas; and Tokyo, Japan. Since June 1964, he has attended Arizona State University under the sponsorship of the Air Force Institute of Technology. He is a member of Alpha Pi Mu, Industrial Engineering Honor Society, and the Institute of Electrical and Electronics Engineers; also, a student member of the American Institute of Industrial Engineers. He is a Captain in the Air Force, married, and has three children.